Autonomous Vehicle Simulation (AVS) Laboratory, University of Colorado

Basilisk Technical Memorandum

Document ID: Basilisk-rwNullSpace

REACTION WHEEL WHEEL SPEED REDUCTION USING THE RW NULL SPACE

Prepared by	H. Schaub

Status: first release module Documentation
Scope/Contents
The module reads in the reaction wheel (RW) control motor torques, and super imposes an additional torque that reduces the wheel speeds. The RW speed control torque array is projected through the RW null space to ensure that the net torque created is always zero. Thus, the addition of this RW null motion torque has no impact on the earlier control torque performance.

Rev	Change Description	By	Date
1.0	Initial Documentation	H. Schaub	2019-02-09

Contents

1 Module Description

1.1 Module Purpose

This module aims to reduce the reaction wheel (RW) speeds by using the null space of the RW array. The input and output messages of `rwNullSpace` are shown in Figure 1. There are three required input messages:

1. The feedback control torque which is mapped onto the RW motor torque solution space. The message type is `RWArrayTorqueIntMsg`, the same message type as the module output message.

2. The RW speed array message of type `RWSpeedIntMsg`.

3. The RW configuration message contains the RW spin axes information. The message type is `RWConstellationFswMsg`.

The module output message is a RW motor torque array message that sums the input control torque solution with the new null space torque solution that will reduce the RW speeds.

1.2 RW Null Space Mathematics

Let N be the number of RWs present, while $\hat{\boldsymbol{g}}_{s_i}$ is the i^{th} RW spin axis unit direction vector. The RW spin axes matrix is defined as

$$[G_s] = [\hat{\boldsymbol{g}}_{s_1} \cdots \hat{\boldsymbol{g}}_{s_N}] \tag{1}$$

The null motion RW projection matrix $[\tau]$ is given by[1]

$$[\tau] = [I_{N\times N}] - [G_s]^T \left([G_s][G_s]^T\right)^{-1} [G_s] \tag{2}$$

This project matrix maps any vector $\boldsymbol{d}$ into the null-space of $[G_s]$ such that no torque is exerted onto the spacecraft. As a result, these null-motion solution never impact the stability or performance of the RW attitude control solution. This concept is illustrated through:

$$\begin{aligned}
[G_s][\tau] &= [G_s]\left([I_{N\times N}] - [G_s]^T\left([G_s][G_s]^T\right)^{-1}[G_s]\right)\\
&= [G_s] - [G_s][G_s]^T\left([G_s][G_s]^T]\right)^{-1}[G_s]\\
&= [G_s] - [G_s]\\
&= [0_{N\times N}]
\end{aligned}$$

1.3 RW Wheel Speed Reduction Control

Let $J_{s_i} > 0$ be the RW inertia about the spin axis $\hat{\boldsymbol{g}}_{s_i}$, Ω_i be the RW spin speed, and $\dot{\boldsymbol{\omega}}$ be the inertial spacecraft angular acceleration vector. The RW motor torque equation is given by[1]

$$u_{s_i} = J_{s_i}(\dot{\Omega}_i + \hat{\boldsymbol{g}}_{s_i}^T\dot{\boldsymbol{\omega}}) \tag{3}$$

Assuming that the spacecraft angular accelerations are much smaller than the wheel accelerations, this is approximated as

$$u_{s_i} \approx J_{s_i}\dot{\Omega}_i \tag{4}$$

Let d_i be the desired torque to reduce the i^{th} RW spin rate Ω_i given through

$$d_i = -K\Omega_i \tag{5}$$

where the feedback gain $K > 0$. Then setting $u_{s_i} = d_i$ ideally provides the stable closed loop response

$$J_{s_i}\dot{\Omega}_i + K\Omega_i = 0 \tag{6}$$

Let $\boldsymbol{d}$ be the $N \times 1$ array of desired RW decelerating motor torques given by

$$\boldsymbol{d} = -K\boldsymbol{\Omega} \tag{7}$$

If this RW motor torque were directly applied then a non-zero torque would be produced onto the spacecraft causing attitude deviations. Instead, this desired despin torque $\boldsymbol{d}$ is mapped through $[\tau]$ onto the null space of the RW array using

$$\boldsymbol{u}_{s,\text{null}} = [\tau]\boldsymbol{d} \tag{8}$$

Assume the attitude feedback RW motor control solution is given by $\boldsymbol{u}_{s,\text{cont}}$, then final module RW motor torque array is the sum of these two torques.

$$\boldsymbol{u}_s = \boldsymbol{u}_{s,\text{cont}} + \boldsymbol{u}_{s,\text{null}} \tag{9}$$

2 Module Functions

This module has the following functions:

- **Evaluate RW null projection matrix** $[\tau]$: When reset the module will pull in the current RW configuration data and create the null motion projection matrix. This matrix remains fixed unit the module is reset again.
- **Compute a RW deceleration torque**: With each update call the module computes a decelerating RW torque solution that lies in the null space of the RW array.
- **Output a net RW motor torque solution**: The module combined the feedback control torque and the null space torque to slow down the RW speeds and outputs a net solution solution.

3 Module Assumptions and Limitations

The module assumes all RW devices are operating and available. It also assumes the RW spin axes don't change during the regular update cycles.

4 Test Description and Success Criteria

The module is run with a fixed RW configuration setup to compare the expected output with the actual BSK module output. The true output is evaluated within the Python script.

4.1 Check 1

In this case only 3 RWs are provided. Here there is no RW null space and the module output should simply be the $\boldsymbol{u}_{s,\text{cont}}$ array.

4.2 Check 2

Here 4 RWs are provided yielding a functional null space to exploit. In this scenario the expected module torque output is evaluated within the Python script.

5 Test Parameters

The following information pertains to the 4 RW case. If only 3 RW are simulated, then the 4th wheel information is omitted. The tests are setup with the following RW spin axis configuration:

$$\hat{\boldsymbol{g}}_{s_1} = {}^{\mathcal{B}}\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \hat{\boldsymbol{g}}_{s_2} = {}^{\mathcal{B}}\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \hat{\boldsymbol{g}}_{s_3} = {}^{\mathcal{B}}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \hat{\boldsymbol{g}}_{s_4} = {}^{\mathcal{B}}\begin{bmatrix} 0.57735 \\ 0.57735 \\ 0.57735 \end{bmatrix} \tag{10}$$

The gain is set to $K = 0.5$, while the RW speeds are

$$\boldsymbol{\Omega} = \begin{bmatrix} 10 & 20 & 30 & 40 \end{bmatrix} \text{ rad/sec} \tag{11}$$

The attitude control torque array is set to

$$\boldsymbol{u}_{s,\text{cont}} = \begin{bmatrix} 0.1 & 0.2 & 0.15 & -0.2 \end{bmatrix} \text{ Nm} \tag{12}$$

The unit testsverify that the module output `RWArrayTorqueIntMsg` messages match expected values.

Table 2: Error tolerance for each test.

Output Value Tested	Tolerated Error
`motorTorque`	1e-06

6 Test Results

The results of the unit test should be included in the documentation. The results can be discussed verbally, but also included as tables and figures.

All of the tests passed:

Table 3: Test results

Check	Pass/Fail
1	PASSED
2	PASSED

7 User Guide

The module must have the feedback gain `OmegaGain` defined. This must be a positive value. Further, all 3 input message connections must be setup.

REFERENCES

[1] Hanspeter Schaub and John L. Junkins. *Analytical Mechanics of Space Systems*. AIAA Education Series, Reston, VA, 4th edition, 2018.