Automatic Balancing and Intelligent Fault Tolerance for a Space-Based Centrifuge

Edward Wilson[*]
Intellization, Redwood Shores, California, 94065

Robert W. Mah[†]
NASA Ames Research Center, Moffett Field, California, 94303

A 2.5 m diameter centrifuge is presently being developed by the Japan Aerospace Exploration Agency (JAXA) for installation on the International Space Station (ISS). While this will enable biological experiments at variable gravity levels, if an imbalance in the large rotating mass of the centrifuge rotor is allowed to persist, it will cause vibrations that disturb the micro-gravity environment of the ISS. This paper presents an approach for automatic balancing and intelligent fault tolerance. It has been developed and tested in simulation on a model of an early centrifuge design prototype developed at NASA Ames Research Center. The high fidelity automatic balancing system (ABS) can sense the imbalance (both static and dynamic) and drive counterweights to minimize the effects of the imbalance. The algorithm consists of an on-line recursive least-squares (RLS) based imbalance estimator which outputs to a simple counterweight control system. The sensor fault detection, isolation, and reconfiguration (FDIR) system uses a maximum likelihood approach. The counterweight FDIR performs a simple check of the encoder and motion commands to detect sticking or skipping. These systems are developed and tested in MATLAB simulation. Extension of these algorithms for application on the present design of the ISS Centrifuge is straightforward and could result in a high performance and autonomously fault-tolerant ABS.

Nomenclature

A – [matrix] a matrix in the general formulation of the LS problem, $Ax \cong b$.

B – [*8-by-1* vector] of strain gauge biases [Newtons]

b – [vector] a vector in the general formulation of the LS problem, $Ax \cong b$.

$\mathrm{c}\psi$ – [scalar] abbreviation for $\cos\psi$. []

E_{NFN} – [*4-by-1* vector] of net force noise – unknown net forces and torques on the rotor [Newtons, Newton-meters]

E_{SFN} – [*8-by-1* vector] of strain gauge force noise – the difference between the F_{net} -generated force and the actual force seen by the strain gauge (e.g., due to vibrations). [Newtons]

E_{SSN} – [*8-by-1* vector] of strain gauge sensor noise – the difference between the actual force seen by the strain gauge and the sensor output. [Newtons]

$F(x_1, x_2, \cdots, x_N)$ – [scalar] joint probability distribution function used in maximum likelihood SG FDIR derivation. []

$f(x_1, x_2, \cdots, x_N)$ – [scalar] joint probability density function used in maximum likelihood SG FDIR derivation. []

F_{net} – [*4-by-1* vector] net force and torque on the rotor, measured in the rotor frame. Equal to $\begin{bmatrix} F_x & F_y & \tau_x & \tau_y \end{bmatrix}^T$. [Newtons, Newton-meters]

---

[*] President, Ed.Wilson@intellization.com, AIAA member
[†] Research Scientist, SSRL Group Lead, Robert.W.Mah@nasa.gov

F_x – [scalar] element of F_{net}, net force on the rotor in the +x in the rotor frame. +y direction indicated by F_y. [Newtons]

G – [8-by-4 matrix] constant matrix that accounts for the strain gauge locations in the transformation from F_{net} to S_{true}. "G" is for "geometry."

g – [scalar] acceleration due to gravity. [meters/second^2]

I – [8-by-8 matrix] identity matrix.

i – [scalar] failure mode number. []

$i_{isolated}$ – [scalar] failure mode number that has been declared as the isolated fault (whether true or not). []

i^* – [scalar] true failure mode number. []

J – [scalar] cost variable minimized in the least squares derivation.

k – [scalar] sample time counter. []

l – [scalar] z-axis coordinate of the upper counterweight plane. Lower plane coordinate is $-l$. The origin of the z-axis is defined as the midpoint between the upper and lower counterweight planes. [meters]

m – [scalar] mass of the rotor. [kg]

m_c – [scalar] mass of the MCI. Can be chosen arbitrarily. [kg]

m_p – [scalar] mass of the PMI. Can be chosen arbitrarily. [kg]

m_{cCW} – [scalar] mass of the mass-couple used to represent the effects of the counterweights. [kg]

m_{pCW} – [scalar] mass of the point mass used to represent the effects of the counterweights. [kg]

N – [scalar] number of measurements in the window used in maximum likelihood SG FDIR. []

$r_{i,k}$ – [scalar] residual corresponding to failure mode i, measured at time update k, used for SG FDIR. [Newtons]

S – [8-by-1 vector] filtered and combined strain gauge force measurement vector. [Newtons]

S_{true} – [8-by-1 vector] the true forces carried by the strain gauge force transducers. [Newtons]

S_{Axf} – [scalar] filtered and combined strain gauge reading indicating the force in the A strain gauge plane (upper) in the +x direction for the *fixed* pair of gauges. B plane, +y direction, *spinning* pairs of gauges indicated by $S_{Ayf}, S_{Bxf}, S_{Byf}, S_{Axs}, S_{Ays}, S_{Bxs}, S_{Bys}$. [Newtons]

$s\psi$ – [scalar] abbreviation for $\sin\psi$. []

T – [8-by-8 matrix] transformation matrix that accounts for the angle of the rotor relative to the fixed gauges. "T" is for "transformation." []

t - [scalar] time. [seconds]

x – [vector] a vector in the general formulation of the LS problem, $Ax \cong b$.

$x_c, y_c, v_{xc}, v_{yc}, a_{xc}, a_{yc}$ – [scalars] position, velocity, and acceleration, in x and y directions, of the MCI. No further subscripts implies the imbalance, the subscript CW indicates the counterweights. [meters, meters/second, meters/second^2]

$x_p, y_p, v_{xp}, v_{yp}, a_{xp}, a_{yp}$ – [scalars] position, velocity, and acceleration, in x and y directions, of the PMI. No further subscripts implies the imbalance, the subscript CW indicates the counterweights. [meters, meters/second, meters/second^2]

W – [diagonal matrix] weighting matrix in the batch LS solution, $\hat{x} = \left(A^T W A\right)^{-1} A^T W b$. []

z_{Af} – [scalar] z-axis coordinate of the S_{Axf} and S_{Ayf} strain gauges. Coordinates of the B plane and *spinning* pairs of gauges indicated by z_{Bf}, z_{As}, z_{Bs}. [meters]

z_m – [scalar] z-axis coordinate of the rotor center of mass. [meters]

$z_{m\phi x}$ – [scalar] linearizing identified variable, $z_{m\phi x} \triangleq z_m \phi_x$. $z_{m\phi y} \triangleq z_m \phi_y$ is the corresponding variable for the y-axis. [meter-radians]

α – [scalar] rotor angular acceleration. [radians/second^2]

β – [scalar] strain gauge bias. [Newtons]

Γ – [4-by-8 matrix] constant matrix that is a function of the fixed system geometry only (strain gauge locations, etc.), used in the estimation of $\hat{F}_{net}$. $\Gamma \triangleq (G^T G)^{-1} G^T$.

γ – [scalar] the generalized likelihood ratio. []

δ – [4-by-1 vector] of effective counterweight coordinates. Positions, velocities, and accelerations are indicated by $\begin{bmatrix} \delta_1 & \delta_2 & \delta_3 & \delta_4 & \dot{\delta}_1 & \dot{\delta}_2 & \dot{\delta}_3 & \dot{\delta}_4 & \ddot{\delta}_1 & \ddot{\delta}_2 & \ddot{\delta}_3 & \ddot{\delta}_4 \end{bmatrix}$. [meters, meters/second, meters/second^2]

ε – [scalar or vector] noise.

θ_{CW} – [4-by-1 to 12-by-1 vector] the vector of counterweight parameters.

$\theta_{CW_{pos,desired}}$ – [4-by-1 vector] the vector of desired counterweight position parameters.

θ_{IB} – [4-by-1 to 12-by-1 vector] the vector of imbalance parameters to be estimated.

$\theta_{IB_{pos}}$ – [4-by-1 vector] the vector of imbalance position parameters.

$\theta_{Misalignment}$ – [4-by-1 vector] the vector of spin-axis misalignment parameters to be estimated.

μ – [scalar] mean of the Gaussian probability distribution used to approximate variability in the SG residuals.

σ – [scalar] standard deviation of the Gaussian probability distribution used to approximate variability in the SG residuals.

τ_x – [scalar] element of F_{net}, net torque on the rotor about the +x axis of the rotor frame, caused by the imbalance. +y direction indicated by τ_y. [Newton-meters]

$\Phi(m_p, m_c)$ – [4-by-4 to 4-by-12 matrix] the Φ matrix from $F_{net} = \Phi \theta$, written as a function of the point mass and mass couple. Can represent either $\Phi_{IB} = \Phi(m_p, m_c)$ or $\Phi_{CW} = \Phi(m_{pCW}, m_{cCW})$.

Φ_{IB} – [4-by-4 to 4-by-12 matrix] the Φ matrix from $F_{net} = \Phi \theta$, for the imbalance.

Φ_{CW} – [4-by-4 to 4-by-12 matrix] the Φ matrix from $F_{net} = \Phi \theta$, for the counterweights.

$\Phi_{Misalignment}$ – [4-by-4 matrix] the Φ matrix from $F_{net} = \Phi \theta$, for the spin-axis misalignment.

ϕ_x – [scalar] spin-axis misalignment angle, about the x-axis. ϕ_y indicates misalignment about the y-axis. [radians]

ψ – [scalar] absolute rotor angle. [radians]

ω – [scalar] rotor angular rate. [radians/second]

Frames: unless otherwise specified calculations are in the rotor frame.

Subscripts: For F_{net} and its elements, no subscript means the total; additional subscripts, as in $F_{netIB}, F_{netCW}, F_{netMisalignment}$, indicate that part of the net force due to the imbalance (IB), counterweights (CW), and axial misalignment (Misalignment). F_{netIB} is broken down further into that part due to the PMI, F_{netIB_p}, and that due to the MCI, F_{netIB_c}.

Estimated/identified values are indicated with a ^. For example, $\hat{F}_{net}$ is the estimate of F_{net}, obtained through a least squares fit of the strain gauge measurements.

Units for each variable are given in [] at the end of each definition. If a variable is unit-less, empty brackets are drawn, if the units are not known exactly, no brackets are drawn.

Abbreviations are listed in Appendix A.

I. Introduction

The autobalancing control systems developed and presented here were developed initially for a centrifuge hardware prototype developed at NASA Ames, and before the ISS Centrifuge design was developed. This hardware prototype is described in detail in this section, along with comparison and explanation of relevance to the ISS Centrifuge design.

A space based centrifuge will allow biological experiments below 1g (the ISS Centrifuge will allow a full range from 0-2g), for example, Mars (0.38g) and Moon (0.16g) levels of gravity, something not possible on Earth. Also, by running the centrifuge at 1g, it serves as a valuable control for comparison with experiments to be run in zero g on the ISS. Previous experiments from previous smaller space-based centrifuges have demonstrated the need for such a control.

However, other experiments aboard the ISS require micro-gravity vibration isolation. So any imbalances in the centrifuge that might cause vibrational disturbances on the ISS must be attenuated by automatic on-line balancing as well as vibration isolation of the centrifuge. The research reported here focuses only on the automatic on-line balancing and the associated fault tolerance. In an automatic balancing system, feedback from sensors measuring rotor motions or forces (e.g., strain gauges, displacement sensors, or accelerometers) is used to drive active counterweights to null the imbalance and resulting vibrations. A supporting system may include a passive or active vibration isolation system to further attenuate any vibrations emanating from the centrifuge rotor.
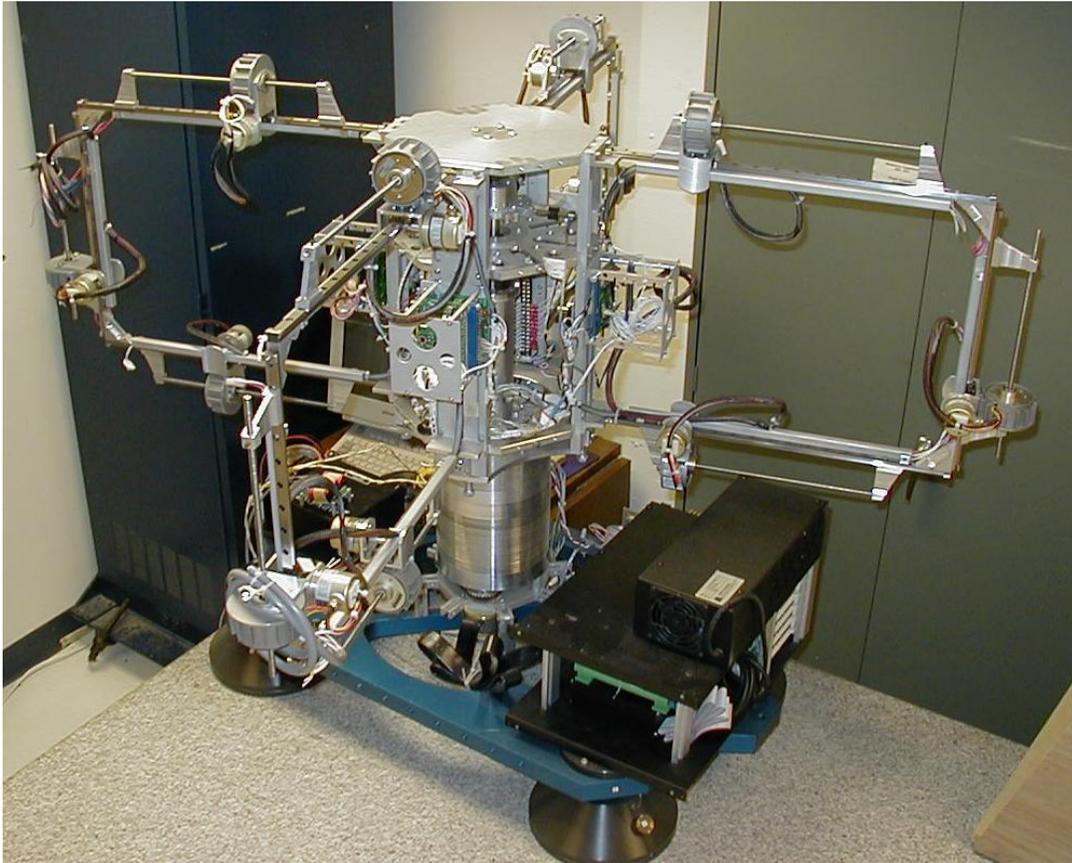
A. SSRL centrifuge laboratory prototype

Figure 1. The SSRL centrifuge laboratory prototype - autobalancing hardware simulator

The NASA Ames Smart Systems Research Laboratory (SSRL) Centrifuge laboratory prototype, developed in 1993-1995, is shown in Figure 1‡. It can float on four 8-inch diameter Fox Air Bearings with spherical air bearing pivots, allowing the vertical spin axis to translate in x and y in response to imbalances. The spin axis is permitted to rotate slightly, as discussed below, but it is a much stiffer spring suspension than the Vibration Isolation Mechanism (VIM) of the present ISS Centrifuge design.

There are 12 counterweights (CWs) constructed of stepper motors (providing the counterbalance mass as well as actuation) and encoders that drive along lead screws (also known as ACME screws or trapezoidal screws–these are different from the ISS Centrifuge which uses ball screws). The CWs are located in upper and lower planes of 4 radial CWs and 4 vertically moving CWs. The vertically moving ones are not used in this research, and the 8 remaining CWs provide redundancy as only 4 are required. The redundant CWs may be used to simulate animal movement or creating disturbances.

There are locations for 24 strain gauges (SGs) to measure imbalance forces. Eight of them do not rotate, and are for measuring the force between the stator and the air bearing platform. No SGs are installed at those 8 locations. 8 SG locations rotate with the rotor, near the upper CW plane. Of these 8 locations, 4 SGs are installed, equally spaced through one rotation (90 degrees apart). Same goes for a plane near the lower CW plane, meaning that the hardware presently has a total of 8 rotating SGs. The SG assemblies are viscoelastic, allowing for slight displacement (on the order of 1-2 mm) and providing damping. The angular (tip-tilt) natural frequency of the rotor is on the order of 3 Hz, and the damping ratio is about 0.1. Planned, but not yet installed, are displacement sensors that would measure the base translations in the horizontal dimensions.

Hardware implementation of this autobalancing technology on the SSRL Centrifuge began in 1996, but was tabled due to lack of funding. The project lay dormant until 2001-2002 when the FDIR capability was added to the MATLAB simulation.

B. SSRL centrifuge research, historical overview

In 1993, the SSRL initiated the development of a space centrifuge testbed. The purpose was to provide an in-house capability to test advanced technologies that may greatly enhance centrifuge control and operation, and to demonstrate proof-of-concept operation. Due to funding cuts in 1995, the centrifuge testbed development effort was put on hold while the advanced technology development work continued at a lower level of effort. The testbed remains intact and can be reconfigured to reflect the current ISS Centrifuge Rotor design.

The initial SSRL goals were focused on developing advanced technologies for "real-time adaptive control of time varying nonlinear systems with unknown structures." The rationale was that the centrifuge rotor properties were time varying and not known well since there are random animal motions, dynamic fluid coupling loads, aerodynamic effects from different habitat configurations and placements, etc. The presumptions were that centrifuge rotor balancing had to be performed in as close to real-time as possible to achieve near-perfect balance condition, and that any residual vibrational disturbances (since it is impossible to completely cancel the effect of a sudden impact) will have to be vibration isolated from the ISS to preserve the microgravity environment required by other experiments on the ISS.

One of the initial SSRL objectives was to determine the optimal configuration (of sensors, counterweights, vibration isolation actuators, and passive vibration isolation elements) for real-time centrifuge rotor balancing and vibration control and isolation. Engineering analyses were performed on various configurations including:

- Counterweights on circular rings about the spin axis

- Counterweights on linear tracks, both parallel to and radial to the spin axis

- Vibration isolation actuators mounted on the oscillating centrifuge base

- Vibration isolation actuators mounted on a stationary barrier that enclosed the oscillating centrifuge base

Other SSRL objectives include developing, testing, and validating the performance of:

- Real-time autobalancing algorithms

- Robust centrifuge operation in face of sensor failures: online fault detection isolation reconfiguration (FDIR) using online neural networks (NN) learning technology

- Sensor noise reduction or cancellation using Adaptive Noise Cancellation technology

- Robust centrifuge operation in face of balancing mechanism failures: detection and isolation of failure, followed by reconfiguration using the remaining counterweights

- Optimum sensor configuration: rotating sensors vs. non-rotating sensors; force vs. displacement vs. acceleration; and combinations of the preceding options

- Active vibration isolation control: using long stroke voice coil actuators mounted on stationary barrier or oscillating centrifuge base (C shaped end-effectors to prevent impact contact between stationary barrier and oscillating centrifuge base)

Initial advanced technology algorithms under development have included:

- Offline/online NN mapping of sensor readings to known imbalance (counterweight compensation is then required at location 180 degrees from identified imbalance; information on individual counterweight positioning is not determined here)

- Offline/online NN pseudo inverse modeling (imbalance location, or sensor readings, to counterweight positions for the balance condition; calibrates counterweight positions for precise static/dynamic balancing)

- Nonlinear restoring force that allows free oscillation of the centrifuge base at the center of an enclosure, re-centers the centrifuge base location, and prevents impact with the enclosure barrier

Advanced technology software solutions under development employed the following:

- Levenberg-Marquardt trained, multiple layer, feed forward neural networks

- Cerebellar Model Articulation Controller (CMAC)

- Radial basis neural networks

- Maximum likelihood, recursive least squares

C. Comparison of SSRL and ISS centrifuges

During this period of technology development at the SSRL, the ISS Centrifuge design was developed by JAXA, along with contractors Toshiba and NT Space, and is now in a near final form[1-5]. Significant differences between the SSRL Centrifuge and the current ISS Centrifuge design are that:

- The spin axis for the SSRL Centrifuge can translate freely in x and y, but not wobble (rotational motion of the spin axis is stiffly constrained – modeled as perfectly stiff for now), whereas the ISS centrifuge spin axis has 3 translational and 2 rotational degrees of freedom, supported by a very compliant vibration isolation mechanism consisting of springs and dampers (both passive and active).

- The imbalance sensors for the SSRL Centrifuge are strain gauges both in the rotor and the stator, measuring the imbalance forces, whereas the ISS Centrifuge uses eddy current displacement sensors to measure the displacements of the compliant vibration isolation mechanism (VIM) resulting from imbalance.

Due to the differences between the two configurations, this research will need to be extended to enable application to the ISS Centrifuge. The following outlines steps towards completing this task.

1. Removing the constraint that the spin axis is fixed, allowing free or passively/actively constrained motion of the spin-axis.

2. Represent imbalance differently, to account for 3-D model – e.g., mass center location and inertia matrix

3. Use the ISS Centrifuge counterweight locations.

4. Use the ISS Centrifuge bearing displacement sensors (BDSs).

5. Model the ISS Centrifuge vibration isolation mechanism (VIM), including springs and dampers, both passive and active.

6. Integrate with ADAMS from MSC Software for testing – thought is to have a simpler, medium fidelity model derived (by hand) for the autobalancer design, but then to test it on the ADAMS model that is fully nonlinear and models the rotor flexibility.

Although the configuration change is significant, it should be possible to extend the basic approaches taken for the SSRL configuration to the ISS configuration, following the overall approach of the autobalancing and FDIR systems.

The key novel concepts in the present design will apply directly to the updated design. These are:
1. Modeling the rotor as a rigid body, and condensing all sensor measurements at each time update to a concise representation of the imbalance-induced forces and moments on the rotor.

2. A control system that follows an indirect adaptive control architecture. It explicitly estimates the imbalance parameters, with the effect of counterweights calculated and subtracted out. Then a straightforward counterweight control system drives the CWs to null the total imbalance.

3. The sensor FDIR system ties in directly with the estimation of imbalance forces and moments, allowing efficient fault detection and isolation, and then reconfiguration by ignoring the failed sensor.

4. The actuator FDIR system ties in directly with the CW control system, allowing hardware redundancy to account for a stuck CW.

5. The combined sensor and actuator FDIR systems should allow autonomous fault tolerance, allowing the system to continue operation in the face of sensor or actuator faults. The segmentation of sensing/identification and actuation/balancing facilitates this reconfiguration, as, for example, a failed actuator does not impact the identification at all.

Brief summary of the autobalancing method presented in this report:
1. The centrifuge rotor is modeled as a rigid body spinning about a fixed axis.

2. The rotor imbalance is represented in a compact, intuitive way as the x- and y-locations of a point mass in a central plane and a pair of asymmetrically located point masses in off-central planes. This four-parameter representation is sufficient to represent an arbitrary imbalance and can be intuitively related to counterweight motions.

3. The rotor imbalance (not including counterweights) is estimated at each sample period as follows:

   a. Sensor signals (strain gauges, counterweight positions, velocities, and accelerations, rotor angle encoder and tachometer) are combined, using a least-squares fit, to calculate the estimated net force and torque in x and y (F_{xIB} F_{yIB} τ_{xIB} τ_{yIB}) created by the imbalance.

   b. The four imbalance parameters (mentioned in (2) above) and their derivatives are estimated using these forces and torques. This uses a dynamic model of the rotor that calculates effects due to the position, velocity, and acceleration of the imbalance parameters.

4. The counterweights are driven to exactly counteract the estimated rotor imbalance.

This autobalancing approach can be considered an indirect method (analogous to indirect vs. direct adaptive control) since the sensor signals are used to build a model of the imbalance, then corrective action is taken based on the identified model parameters. In a direct method, (filtered, and mathematically manipulated) sensor signals would be used to directly drive the counterweights. Hopefully, this feedback loop would drive the counterweights until the sensors read zero. The increased complexity of the indirect method presented here enables more accurate fitting of the sensor data to the dynamic model of the imbalance. Whether the increased accuracy of the indirect method produces results that are sufficiently better than those of the direct method will depend on the characteristics of the imbalance (how fast it is moving, etc.) and sensor noise.

A previous similar approach was developed by the author, concluding in October 1995. The major improvement made in the current version involves breaking up the identification so that all sensor signals are reduced (combined) to result in an intermediate estimation of the net forces and torques on the rotor. In the previous approach, all sensors were used directly to identify the imbalance parameters. In the present approach, all sensors (strain gauges,

counterweight positions, velocities, and accelerations, rotor angle encoder and tachometer) are used at each sample period to calculate four variables: the estimated net force and torque in x and y (F_{xIB} F_{yIB} τ_{xIB} τ_{yIB}) created by the imbalance. This process involves a least squares fit to the data, using a model of the sensor geometry and subtracting out known forces due to the counterweights. These four variables then pass to the imbalance identification algorithm. Benefits of segmenting the identification into these two parts are:

1. Physically, the four imbalance parameters are directly related to these four intermediate variables (F_{xIB} F_{yIB} τ_{xIB} τ_{yIB}). The relation between sensor values and these intermediate variables is more direct than that between sensor values and imbalance parameters. This logically separates estimation of forces and torques created by the imbalance from the estimation of imbalance parameters themselves.

2. It is easier to identify failed sensors, since the analysis can be performed without regard to the imbalance dynamics – one can analyze the residuals in the estimation of the intermediate variables.

3. If sensors change (e.g., on-line failure, design change, etc.), the second part of the identification (that finds imbalance parameters from estimated forces and torques) does not have to be changed.

4. Overall complexity is reduced by breaking one large problem into two smaller ones. No accuracy is lost, due to the physical reasoning listed in (1) above.

D. Related research

Extensive literature exists on balancing rotating machinery, including centrifuges. Some of this literature relates to on-line automatic balancing, but there is very little published work related to balancing of a space-based centrifuge. The micro-gravity dynamics, extreme sensitivity to vibrations, and requirement to track moving imbalances (due to moving rodents in the centrifuge habitats) puts this research outside the region of relevance of much of the existing work.

The developers of the actual ISS centrifuge autobalancing system at Toshiba, NEC Toshiba Space Systems, and JAXA have published summaries of their control systems[1-5], which should be taken as the state of the art in this field. In any well designed control system there is an important balance between performance and complexity. Although direct comparison of the control system approach presented here with the ISS Centrifuge control system design has not been performed, it is judged that the authors' approach will provide higher performance (faster response to imbalance changes and reduced vibration generation) and better fault tolerance at the expense of complexity of software implementation and support.

E. Reader's guide

Section II briefly describes the control system architecture.

Section III describes the imbalance parameter estimation in extensive detail. Many readers will prefer to skim this section, as it provides in-depth documentation of the various permutations of imbalance identification algorithms. Sensor FDIR is included here.

Section IV describes the algorithms to calculate CW commands based on the identification results, along with CW FDIR.

Section V describes the MATLAB software simulation implementation, along with example results.

In this report, equations are numbered. Repeated equations share the number of the original equation, with the numbering italicized.
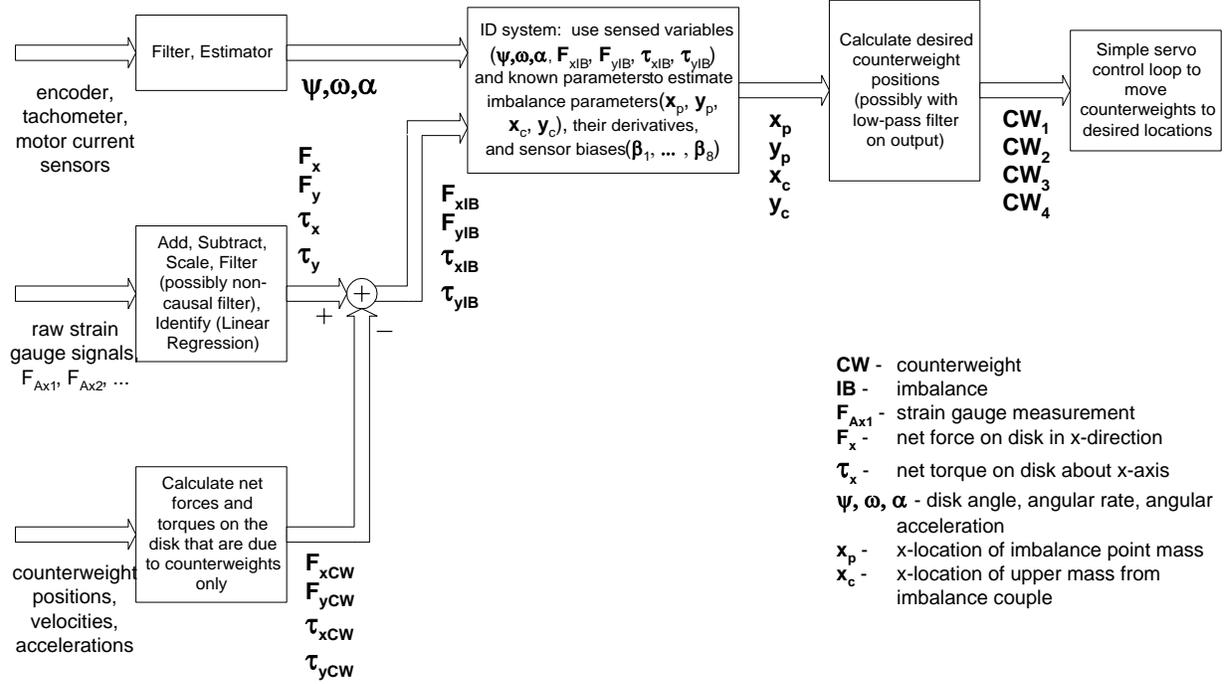
II. Control system architecture

Figure 2. Simplified control system block diagram

The control system architecture is presented in Figure 2. At the highest level the controller works in two steps: 1. the imbalance parameters are estimated (x_p, y_p, x_c, y_c); and 2. the counterweight locations that null this identified imbalance are calculated and commanded.

The imbalance parameter estimation first reduces all sensor inputs into a vector of net forces and torques on the rotor $(F_x, F_y, \tau_x, \tau_y)$, subtracts out the force vector contributed by the CWs $(F_{xCW}, F_{yCW}, \tau_{xCW}, \tau_{yCW})$, leaving the force vector attributable to the imbalance of the rotor itself, without CWs $(F_{xIB}, F_{yIB}, \tau_{xIB}, \tau_{yIB})$. This vector (note that these are not physical vectors) is then used, along with a model of the centrifuge rotor, to identify the rotor imbalance parameters.

Details of the identification are presented in Section III, with sensor FDIR presented in Section III.L, and details of the counterweight control and FDIR are presented in Section IV.

III. Imbalance identification

The basic approach taken here to identify the imbalance, or to estimate its parameters, is:

1. Calculate (identify) the net forces and torques on the rotor at each sample period, based upon the strain gauge measurements.
2. Subtract out the forces due to the counterweights, leaving the forces due only to the imbalances.
3. Identify the "imbalance parameters" corresponding to these forces that define the state of imbalance in the rotor.

An indirect, model-based approach like this should work well if the form of the model can be identified correctly and the sensors are not excessively noisy or biased.

This section contains very detailed derivations of the imbalance parameter estimation algorithms, for a variety of configuration options. However, it is important to note that the algorithms have been optimized carefully for run-time efficiency, and they reduce in the end to simple and straightforward matrix operations that are amenable to real-time implementation. The simplified algorithms are summarized in Section III.J.

The sensor FDIR system, which is tightly integrated with the imbalance parameter estimation, is described in Section III.L.

A. Least squares regression background

Both the reduction of strain gauge measurements to net forces and torques, and the identification of imbalance parameters are achieved using least squares regression. The mathematics behind this approach is reviewed briefly here. The standard form for a linear least squares problem is given in Eq. (1) or (2)

$$Ax = b + \varepsilon \tag{1}$$

$$Ax \cong b \tag{2}$$

where b is a vector of (perfect) measurements, ε is a vector of measurement noise, x contains the parameters to be identified, and matrix A contains known variables system parameter values (i.e., A is noise-free). The $\cong$ in the $Ax \cong b$ representation indicates that the left and right sides of the equation would be equal if noise were not present[6]. The LS ID solution, $\hat{x}$, minimizes the sum of the squares of the elements of the error, $A\hat{x} - b$. If the problem at hand can be put into the form of (1), with noise appearing only in the ε term, $\hat{x}$ can be solved directly (i.e., this is a closed-form solution, rather than an iterative optimization as might be required if the equations can not be put into this standard form) using one of the following approaches[6-8].

$$\text{unweighted, batch algorithm: } \hat{x} = \left(A^T A \right)^{-1} A^T b \tag{3}$$

$$\text{weighted, batch algorithm: } \hat{x} = \left(A^T W A \right)^{-1} A^T W b \tag{4}$$

where W is a diagonal weighting matrix.

Either of these algorithms can be made recursive, and the weighting matrix, W, can be chosen to weight the data according to an exponentially decaying function – as is commonly done when implemented recursively. The recursive and batch solutions are identical since they minimize the same cost function.

In practice, many times the governing equations do not immediately fit exactly the form $Ax = b + \varepsilon$, with, for example, noise being present in the A matrix and the x not being immediately and linearly separated from A and b as required. So the basic approach to LS ID is to find some governing equations (the equations of motion, for example) that contain the parameter values to be identified and measurement data. Then these equations are manipulated to conform to the $Ax \cong b$ formulation, possibly requiring approximations along the way (dropping higher order terms, for example).

B. Estimation of the net forces and torques using strain gauge signals

The goal in this section is to develop a procedure to produce net force and torque estimates, $\hat{F}_{net}$, at each sample period based upon strain gauge measurements. $\hat{F}_{net}$ contains the net force and torque acting on the rotor, measured in the (rotating) rotor frame. There will be between four and eight (and possibly more) strain gauge measurements used to produce the four net force and torque signals. Strain gauges may be spinning with the rotor or fixed in the base. Each strain gauge will have a bias, which must be accounted for. The approach taken here is to solve a least-squares fit at each sample to get the net forces and torques, while keeping the bias terms separate so that they can be identified in the imbalance identification step.

For now, assume the following strain gauge layout. There are 16 strain gauges: 8 fixed, 8 spinning. They are arranged in sets of four. The upper set of fixed strain gauges is shown in the following sketch. "S" indicates "strain gauge." "A" or "B" (not shown here) indicates the upper or lower set. "x" or "y" indicates the axis of measurement. "f" or "s" (not shown here) indicates "fixed" or "spinning" gauges. "1" or "2" identify gauges on the same axis. "1" is on the positive side of the axis and "2" is on the negative side of the axis. When there is a force on the rotor in the positive direction, "1" will be in tension and "2" will be in compression.
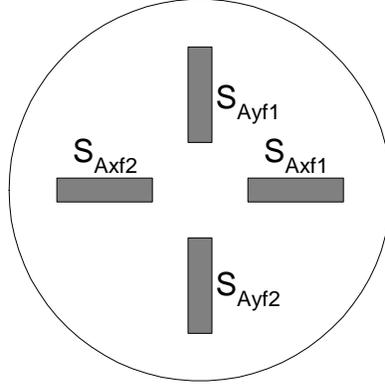
Figure 3. Strain gauge layout

The first step is to combine two strain gauges on the same axis, which should yield exactly opposite readings. To improve accuracy, these readings will combined by subtraction. For example, $S_{Axf} = (S_{Axf1} - S_{Axf2})/2$. If a force of +10 Newtons is applied to the rotor in the +x direction, S_{Axf1} will be in tension, reading +10 N and S_{Axf2} will be in compression, reading -10 N. The combined value, S_{Axf}, will be (10 - (-10))/2 = 10 N. This combined value will also be filtered to reduce sensor noise and extraneous vibrations. For now, this operation is represented as filter( ).

$$S_{Axf} = \text{filter}\big((S_{Axf1} - S_{Axf2})/2\big)$$
$$S_{Ayf} = \text{filter}\big((S_{Ayf1} - S_{Ayf2})/2\big)$$
$$S_{Bxf} = \text{filter}\big((S_{Bxf1} - S_{Bxf2})/2\big)$$
$$S_{Byf} = \text{filter}\big((S_{Byf1} - S_{Byf2})/2\big)$$
$$S_{Axs} = \text{filter}\big((S_{Axs1} - S_{Axs2})/2\big) \tag{5}$$
$$S_{Ays} = \text{filter}\big((S_{Ays1} - S_{Ays2})/2\big)$$
$$S_{Bxs} = \text{filter}\big((S_{Bxs1} - S_{Bxs2})/2\big)$$
$$S_{Bys} = \text{filter}\big((S_{Bys1} - S_{Bys2})/2\big)$$

Assume that this force measurement is composed of three parts:
$$S = S_{true} + \beta + \varepsilon \tag{6}$$

1) S_{true}, the actual force transmitted by the combined pair of strain gauges (if the sensor were perfect, $S = S_{true}$). However, S_{true} may contain force disturbances that are not due to imbalances (such as structural vibrations).

2) β, a bias term that represents the net bias of the combined pair of strain gauges (assumed to be non-time-varying). With eight strain gauge pairs, there will be eight biases.

3) ε, sensor noise, an unbiased white noise signal due to sensor error.

Putting this in vector form:
$$S = S_{true} + B + E_{SSN} \tag{7}$$

$$
S = \begin{bmatrix} S_{Axf} \\ S_{Ayf} \\ S_{Bxf} \\ S_{Byf} \\ S_{Axs} \\ S_{Ays} \\ S_{Bxs} \\ S_{Bys} \end{bmatrix} ; S_{true} = \begin{bmatrix} S_{Axf_{true}} \\ S_{Ayf_{true}} \\ S_{Bxf_{true}} \\ S_{Byf_{true}} \\ S_{Axs_{true}} \\ S_{Ays_{true}} \\ S_{Bxs_{true}} \\ S_{Bys_{true}} \end{bmatrix} ; B = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \\ \beta_6 \\ \beta_7 \\ \beta_8 \end{bmatrix} ; E_{SSN} = \begin{bmatrix} \varepsilon_{SSN1} \\ \varepsilon_{SSN2} \\ \varepsilon_{SSN3} \\ \varepsilon_{SSN4} \\ \varepsilon_{SSN5} \\ \varepsilon_{SSN6} \\ \varepsilon_{SSN7} \\ \varepsilon_{SSN8} \end{bmatrix}
\tag{8}
$$

The "SSN" subscript in E_{SSN} indicates Strain gauge Sensor Noise.

Since S_{true}, the actual force at the strain gauge, can be directly calculated from the net forces and torques on the rotor, accounting for the rotation transformation and assuming random force noise (actual forces at the strain gauges, such as vibrations, that do not result in a net force or torque on the rotor), it can be put in equation form:

$$
S_{true} = TGF_{net} + E_{SFN} ,
\tag{9}
$$

where,

T is an 8x8 transformation matrix that accounts for the angle of the rotor relative to the fixed gauges. It is a calculable direct function of ψ. The upper right and lower left quadrants are all zeros. The lower right quadrant is the identity matrix (since the spinning gauges do not require transformation). "T" is for "transformation."

G is a constant 8x4 matrix that accounts for the strain gauge locations in the transformation from F_{net} to S_{true}. It contains elements such as $\dfrac{-z_{Bf}}{z_{Af} - z_{Bf}}$. "G" is for "geometry."

$F_{net} = [F_x \quad F_y \quad \tau_x \quad \tau_y]^T$ is the vector of net forces and torques on the rotor, due to all causes (IB, CW, axis).

$E_{SFN} = [\varepsilon_{SFN1} \quad \varepsilon_{SFN2} \quad \varepsilon_{SFN3} \quad \varepsilon_{SFN4} \quad \varepsilon_{SFN5} \quad \varepsilon_{SFN6} \quad \varepsilon_{SFN7} \quad \varepsilon_{SFN8}]^T$, where "SFN" indicates "Strain gauge Force Noise."

To derive G, first assume the rotor angle, $\psi = 0$, so $T = I$, the identity matrix. In this case,

$$
S_{true} = GF_{net}
\tag{10}
$$

$$\begin{bmatrix} S_{Axf_{true}} \\ S_{Ayf_{true}} \\ S_{Bxf_{true}} \\ S_{Byf_{true}} \\ S_{Axs_{true}} \\ S_{Ays_{true}} \\ S_{Bxs_{true}} \\ S_{Bys_{true}} \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} & G_{14} \\ G_{21} & G_{22} & G_{23} & G_{24} \\ G_{31} & G_{32} & G_{33} & G_{34} \\ G_{41} & G_{42} & G_{43} & G_{44} \\ G_{51} & G_{52} & G_{53} & G_{54} \\ G_{61} & G_{62} & G_{63} & G_{64} \\ G_{71} & G_{72} & G_{73} & G_{74} \\ G_{81} & G_{82} & G_{83} & G_{84} \end{bmatrix} \begin{bmatrix} F_x \\ F_y \\ \tau_x \\ \tau_y \end{bmatrix} \tag{11}$$

Each of the elements in G can now be identified by force and torque balance equations, where the SG geometry is specified as shown in Figure 3 and Figure 5.

$$F_x = S_{Axf_{true}} + S_{Bxf_{true}}$$
$$\tau_y = S_{Axf_{true}} z_{Af} + S_{Bxf_{true}} z_{Bf} \tag{12}$$

Solving these two equations with two unknowns,

$$S_{Axf_{true}} = \frac{-z_{Bf}}{z_{Af} - z_{Bf}} F_x + \frac{1}{z_{Af} - z_{Bf}} \tau_y$$

$$S_{Bxf_{true}} = \frac{z_{Af}}{z_{Af} - z_{Bf}} F_x + \frac{-1}{z_{Af} - z_{Bf}} \tau_y \tag{13}$$

Repeating for y,

$$F_y = S_{Ayf_{true}} + S_{Byf_{true}}$$
$$\tau_x = -S_{Ayf_{true}} z_{Af} - S_{Byf_{true}} z_{Bf} \tag{14}$$

$$S_{Ayf_{true}} = \frac{-z_{Bf}}{z_{Af} - z_{Bf}} F_y + \frac{-1}{z_{Af} - z_{Bf}} \tau_x$$

$$S_{Byf_{true}} = \frac{z_{Af}}{z_{Af} - z_{Bf}} F_y + \frac{1}{z_{Af} - z_{Bf}} \tau_x \tag{15}$$

With $\psi = 0$, the only difference for the spinning gauges is the different z-axis gauge locations,

$$S_{Axs_{true}} = \frac{-z_{Bs}}{z_{As} - z_{Bs}} F_x + \frac{1}{z_{As} - z_{Bs}} \tau_y$$

$$S_{Bxs_{true}} = \frac{z_{As}}{z_{As} - z_{Bs}} F_x + \frac{-1}{z_{As} - z_{Bs}} \tau_y$$

$$S_{Ays_{true}} = \frac{-z_{Bs}}{z_{As} - z_{Bs}} F_y + \frac{-1}{z_{As} - z_{Bs}} \tau_x$$

$$S_{Bys_{true}} = \frac{z_{As}}{z_{As} - z_{Bs}} F_y + \frac{1}{z_{As} - z_{Bs}} \tau_x \tag{16}$$

So the G matrix is:

$$G = \begin{bmatrix} \dfrac{-z_{Bf}}{z_{Af} - z_{Bf}} & 0 & 0 & \dfrac{1}{z_{Af} - z_{Bf}} \\[2.5ex] 0 & \dfrac{-z_{Bf}}{z_{Af} - z_{Bf}} & \dfrac{-1}{z_{Af} - z_{Bf}} & 0 \\[2.5ex] \dfrac{z_{Af}}{z_{Af} - z_{Bf}} & 0 & 0 & \dfrac{-1}{z_{Af} - z_{Bf}} \\[2.5ex] 0 & \dfrac{z_{Af}}{z_{Af} - z_{Bf}} & \dfrac{1}{z_{Af} - z_{Bf}} & 0 \\[2.5ex] \dfrac{-z_{Bs}}{z_{As} - z_{Bs}} & 0 & 0 & \dfrac{1}{z_{As} - z_{Bs}} \\[2.5ex] 0 & \dfrac{-z_{Bs}}{z_{As} - z_{Bs}} & \dfrac{-1}{z_{As} - z_{Bs}} & 0 \\[2.5ex] \dfrac{z_{As}}{z_{As} - z_{Bs}} & 0 & 0 & \dfrac{-1}{z_{As} - z_{Bs}} \\[2.5ex] 0 & \dfrac{z_{As}}{z_{As} - z_{Bs}} & \dfrac{1}{z_{As} - z_{Bs}} & 0 \end{bmatrix} \tag{17}$$

Now allowing for $\psi \neq 0$, find the transformation matrix T that performs,

$$S_{true} = T S_{true}\big|_{\psi=0} \tag{18}$$

The spinning gauge forces are independent of the rotor angle, so the lower right quadrant is a 4x4 identity matrix. There is no coupling between spinning and fixed gauge forces, so the upper right and lower left quadrants are all zeros. The upper left quadrant performs the following transformation, where the following abbreviations are made for cos and sin, as will be done throughout this document.

$$c\psi \triangleq \cos\psi$$
$$s\psi \triangleq \sin\psi \tag{19}$$

$$S_{true} = \begin{bmatrix} R(\psi) & 0 \\ 0 & R(\psi) \end{bmatrix} S_{true}\big|_{\psi=0} \tag{20}$$

$$\begin{bmatrix} S_{Axf_{true}} \\ S_{Ayf_{true}} \\ S_{Bxf_{true}} \\ S_{Byf_{true}} \end{bmatrix} = \begin{bmatrix} c\psi & -s\psi & 0 & 0 \\ s\psi & c\psi & 0 & 0 \\ 0 & 0 & c\psi & -s\psi \\ 0 & 0 & s\psi & c\psi \end{bmatrix} \begin{bmatrix} S_{Axf_{true}}\big|_{\psi=0} \\ S_{Ayf_{true}}\big|_{\psi=0} \\ S_{Bxf_{true}}\big|_{\psi=0} \\ S_{Byf_{true}}\big|_{\psi=0} \end{bmatrix} \tag{21}$$

So the T matrix is,

$$T = \begin{bmatrix} c\psi & -s\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ s\psi & c\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c\psi & -s\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & s\psi & c\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{22}$$

T^{-1} will be needed at each sample, so rather than calculate it numerically at run-time it is calculated analytically here. At each sample $\cos\psi$ and $\sin\psi$ need to be calculated once only.

$$T^{-1} = \begin{bmatrix} c\psi & s\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ -s\psi & c\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c\psi & s\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & -s\psi & c\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{23}$$

Now that G and T have been derived, Eq. (7) and (9) are repeated here and combined to yield,

$$S = S_{true} + B + E_{SSN} \tag{7}$$

$$S_{true} = TGF_{net} + E_{SFN} \tag{9}$$

$$S = TGF_{net} + B + E_{SFN} + E_{SSN} \tag{24}$$

Rearranging,

$$S - B = TGF_{net} + (E_{SFN} + E_{SSN}) \tag{25}$$

$$T^{-1}(S - B) = GF_{net} + T^{-1}(E_{SFN} + E_{SSN}) \tag{26}$$

$$GF_{net} = T^{-1}(S - B) - T^{-1}(E_{SFN} + E_{SSN}) \tag{27}$$

This is now in the standard form for a least squares problem ("regression form"): $Ax = b + \varepsilon$, where

$$\begin{aligned} A &= G \\ x &= F_{net} \\ b &= T^{-1}(S - B) \\ \varepsilon &= -T^{-1}(E_{SFN} + E_{SSN}) \end{aligned} \tag{28}$$

The least-squares solution for this equation is $\hat{x} = \left(A^T A\right)^{-1} A^T b$. The representation in Eq. (25), with $(S - B)$ as b and TG as A, could have been used; however, pre-multiplying by T^{-1} keeps the A matrix constant, so it

(and $\left(A^T A\right)^{-1} A^T$) does not need to be re-calculated at each update. The least-squares minimization will be derived here for this problem.

Least squares problem statement

Assume a system governed by the above equation. T^{-1} and G are known.[§] S is measured, but is corrupted by the bias and noise terms as shown. Find $\hat{F}_{net}$ that can reproduce $T^{-1}(S - B)$ by the equation $T^{-1}(S - B) = G\hat{F}_{net}$, where the error $\left(G\hat{F}_{net} - T^{-1}(S - B)\right)^T \left(G\hat{F}_{net} - T^{-1}(S - B)\right)$ is minimized. The idea is that by finding $\hat{F}_{net}$ that minimizes this quadratic cost function will be close to the actual F_{net}, and this problem formulation is mathematically easy to solve, as quickly derived below.

$$\min_{\hat{F}_{net}} \quad J = \left(G\hat{F}_{net} - T^{-1}(S - B)\right)^T \left(G\hat{F}_{net} - T^{-1}(S - B)\right) \tag{29}$$

The cost J is minimized when $\dfrac{\partial J}{\partial \hat{F}_{net}} = 0$.

$$\frac{\partial J}{\partial \hat{F}_{net}} = 0 = 2\frac{\partial}{\partial \hat{F}_{net}}\left(G\hat{F}_{net} - T^{-1}(S - B)\right)^T \left(G\hat{F}_{net} - T^{-1}(S - B)\right)$$

$$0 = 2G^T\left(G\hat{F}_{net} - T^{-1}(S - B)\right)$$

$$0 = G^T G\hat{F}_{net} - G^T T^{-1}(S - B) \tag{30}$$

$$G^T G\hat{F}_{net} = G^T T^{-1}(S - B)$$

$$\left(G^T G\right)^{-1}\left(G^T G\right)\hat{F}_{net} = \left(G^T G\right)^{-1} G^T T^{-1}(S - B)$$

$$\hat{F}_{net} = \left(\left(G^T G\right)^{-1} G^T\right)\left(T^{-1}(S - B)\right) \tag{31}$$

Summarizing, Eq. (32) shows how the filtered strain gauge signals can be processed, along with estimated SG biases, to produce the estimated net force and torque on the rotor at any given sample time.

$$\hat{F}_{net} = \Gamma T^{-1}(S - B) \tag{32}$$

$$\hat{F}_{net} \triangleq \begin{bmatrix} \hat{F}_x \\ \hat{F}_y \\ \hat{\tau}_x \\ \hat{\tau}_y \end{bmatrix} \tag{33}$$

[§] T^{-1} is based on measurements of ψ, but this is highly accurate compared with other measurements. G is based upon measurement of the strain gauge locations, which is assumed to be highly accurate. Additionally, the assumption that the system is governed by the given equation is an important one. Every effort has been made to account for all effects in the model, such as axis misalignment and sensor biases, but there are sure to be some effects that remain unaccounted for.

$$\Gamma \triangleq (G^T G)^{-1} G^T = \begin{bmatrix} \Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\ 0 & \Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} \\ 0 & \Gamma_{32} & 0 & \Gamma_{34} & 0 & \Gamma_{36} & 0 & \Gamma_{38} \\ -\Gamma_{32} & 0 & -\Gamma_{34} & 0 & -\Gamma_{36} & 0 & -\Gamma_{38} & 0 \end{bmatrix} \tag{34}$$

$$T^{-1} = \begin{bmatrix} c\psi & s\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ -s\psi & c\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c\psi & s\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & -s\psi & c\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, S \triangleq \begin{bmatrix} S_{Axf} \\ S_{Ayf} \\ S_{Bxf} \\ S_{Byf} \\ S_{Axs} \\ S_{Ays} \\ S_{Bxs} \\ S_{Bys} \end{bmatrix}, B \triangleq \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \\ \beta_6 \\ \beta_7 \\ \beta_8 \end{bmatrix} \tag{35}$$

The term, $(G^T G)^{-1} G^T$ is a constant 4x8 matrix that is a function of the fixed system geometry only (strain gauge locations, etc.). It is calculated once only, and renamed Γ, where $\Gamma \triangleq (G^T G)^{-1} G^T$. Due to symmetries, some of the elements of Γ are always zero, and other elements are duplicates. There are only 8 independent elements, as shown in Eq. (34).

T^{-1} is a function of ψ, and is calculated at each sample. This is an 8x8 matrix, but the inverse is performed analytically, so only eight terms need to be updated at each sample. There are only two independent elements.

S is the 8x1 vector of measurements resulting from the filtering and combination of strain gauge pairs, as shown in Eq. (8).

B is the 8x1 bias vector. In the calculation, the estimated value, $\hat{B}$, is used since B is not directly known. $\hat{B}$ is calculated in Section III.I.4.

C. Parametric structure to model a general state of imbalance in the rotor

A general form for the imbalance that is capable of representing any possible imbalance is required. An x-y-z point mass location combined with a 3x3 inertia matrix describing the spun portion of the centrifuge (the "rotor") could be used, but a more compact[**] and intuitive representation is presented here. This arbitrarily chosen imbalance model structure has been proven to be sufficiently general to model any possible imbalance.[††]

The form chosen contains two specific perturbations to a perfectly balanced rotor. These "model imbalances" are:

1) A "point-mass" imbalance (PMI) located on the plane equidistant from the two counterweight planes.

2) A "mass-couple" imbalance (MCI) composed of two equal masses located symmetrically about the center of the coordinate system. Each mass is in one of the planes containing the counterweights.

It is not possible to represent a general state of imbalance with only a single point mass located somewhere within the rotor. This is most easily demonstrated by considering a perfect rotor that has two equal masses added to it symmetrically about the mass center (similar to the "mass-couple" imbalance above). This situation can not be represented by a single point mass imbalance, proving that it is not a general representation.

[**] The structure presented is minimal (4 parameters), while still representing all mass properties that matter. For the case of a free spin axis, as with the current ISS Centrifuge design, this set is no longer sufficient.
[††]See Appendix C.

The forces acting on each of these model imbalances (PMI, MCI) are due to gravity, rotor motion (centrifugal acceleration and angular acceleration), and imbalance motion (Coriolis effects, imbalance acceleration)[‡‡]. The imbalance-induced net forces and torques $F_{netIB} \triangleq [F_{xIB}, F_{yIB}, \tau_{xIB}, \tau_{yIB}]^T$ on the rotor will be calculated as functions of the rotor motion (ψ, ω, α), fixed system parameters (gravity, g, and rotor geometry), and imbalance parameters. Once measurements of actual net forces torques and rotor motions are available, it will be possible to estimate the imbalance parameters.
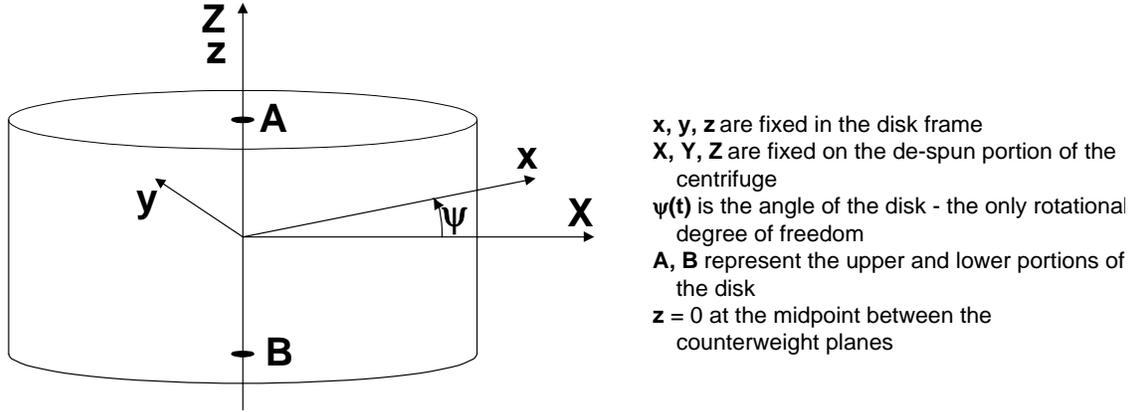

x, y, z are fixed in the disk frame
X, Y, Z are fixed on the de-spun portion of the centrifuge
ψ(t) is the angle of the disk - the only rotational degree of freedom
A, B represent the upper and lower portions of the disk
z = 0 at the midpoint between the counterweight planes

Figure 4. Coordinate system used to describe imbalances and their effects

The coordinate system is shown in Figure 4, with the z-axis locations shown in detail in Figure 5. An xyz Cartesian coordinate system is used to describe imbalance locations fixed within the rotor. The rotor angle, $\psi(t)$, describes the relative angular position of the rotor with respect to the non-spinning (i.e. "de-spun") portion of the centrifuge[§§]. The rotor angular velocity, $\omega(t)$, and angular acceleration, $\alpha(t)$, are the first and second derivatives of $\psi(t)$ with respect to time.
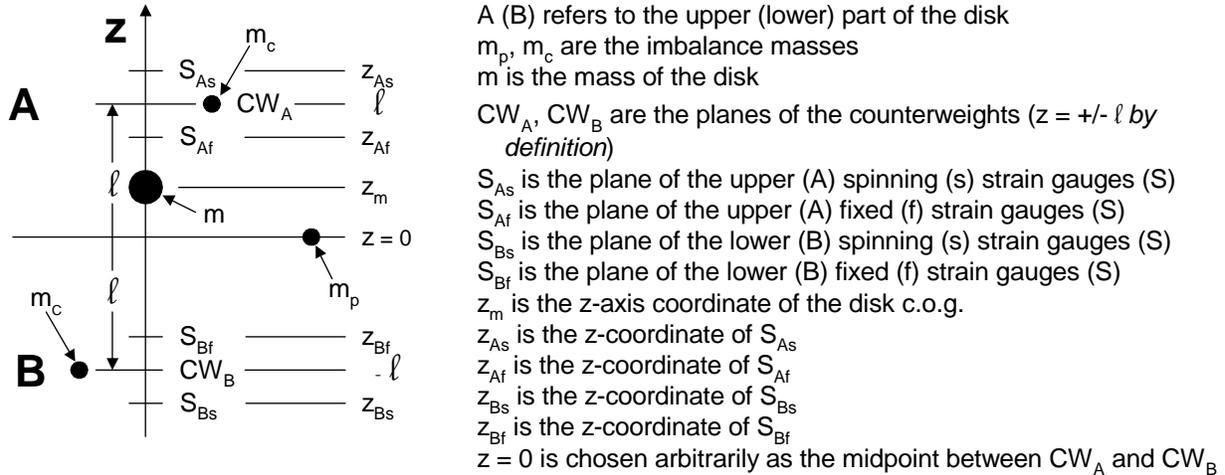

A (B) refers to the upper (lower) part of the disk
m_p, m_c are the imbalance masses
m is the mass of the disk
CW_A, CW_B are the planes of the counterweights (z = +/- ℓ by definition)
S_{As} is the plane of the upper (A) spinning (s) strain gauges (S)
S_{Af} is the plane of the upper (A) fixed (f) strain gauges (S)
S_{Bs} is the plane of the lower (B) spinning (s) strain gauges (S)
S_{Bf} is the plane of the lower (B) fixed (f) strain gauges (S)
z_m is the z-axis coordinate of the disk c.o.g.
z_{As} is the z-coordinate of S_{As}
z_{Af} is the z-coordinate of S_{Af}
z_{Bs} is the z-coordinate of S_{Bs}
z_{Bf} is the z-coordinate of S_{Bf}
z = 0 is chosen arbitrarily as the midpoint between CW_A and CW_B

Figure 5. Z-axis locations of strain gauges, counterweights, and imbalances

The origin of the xyz coordinate system is fixed on the axis of rotation at the midpoint between the upper (A) and lower (B) counterweight planes. The "mass-couple" imbalances, shown in Figure 5, are defined arbitrarily to lie in the same planes as the counterweights. The "point-mass" imbalance is chosen arbitrarily to lie in the plane

---

[‡‡]The rotor angular acceleration force, Coriolis force, and force due to imbalance acceleration are expected to be minimal, but will be included for completeness.
[§§]Although the centrifuge base may actually rotate slightly, this will be a small effect, and it is neglected in this analysis.

equidistant between these two planes. The z-axis coordinate of this plane is defined to be $z = 0$. The z-axis coordinates of the upper and lower counterweight planes are defined to be $z = l$ and $z = -l$.

Two types of strain gauges will be used on both the upper and lower parts of the rotor. "Fixed" gauges (S_{Af} and S_{Bf}) will be attached to the non-spinning base, and "spinning" gauges (S_{As} and S_{Bs}) will be mounted on the spinning rotor. As their z-axis coordinates may not be symmetric, each of the four strain gauge locations has its own coordinate label, $z_{As}, z_{Af}, z_{Bs}, z_{Bf}$, as shown in Figure 5.

To summarize, the physical locations of the counterweight planes define the locations of the z-axis and the model imbalances (PMI and MCI).
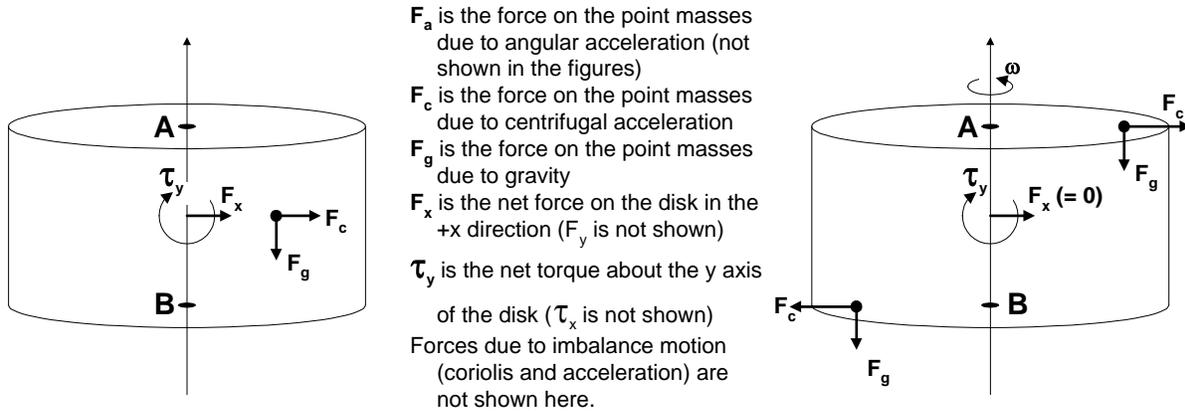

F_a is the force on the point masses due to angular acceleration (not shown in the figures)
F_c is the force on the point masses due to centrifugal acceleration
F_g is the force on the point masses due to gravity
F_x is the net force on the disk in the +x direction (F_y is not shown)
τ_y is the net torque about the y axis of the disk (τ_x is not shown)
Forces due to imbalance motion (coriolis and acceleration) are not shown here.

Figure 6. Model imbalances

Point-mass imbalance (PMI), Mass-couple imbalance (MCI), and their effects on the rotor.

Figure 6 shows the model imbalances used to describe the general state of imbalance in the rotor. These discrete point masses, if placed in the correct locations, can produce the same effects as any imbalance in the rotor. The proof for this assumes that any state of imbalance can be described by a (possibly large) number of point masses of varying mass and location. The net forces and torques due to each of these many masses is calculated, summed, and shown to be equal to the net forces and torques due to the two model imbalances (PMI, MCI), provided the imbalance parameters are chosen correctly (for example, $m_p x_p = \sum_{i=1}^{n} m_i x_i$, where m_i and x_i represent the mass and x-coordinate of each of the many point masses). The effective net forces and torques on the rotor due to the model imbalances will be calculated next.

D. Net forces and torques due to the model imbalance

Even though the spin axis is not perfectly rigid, it is modeled as such in this analysis.

1. Point-mass imbalance

The first model imbalance, the point-mass imbalance, has a mass equal to m_p and is located in the x-y plane, with $z = 0$. The mass magnitude, x-y location, x-y velocity, and x-y-z acceleration all affect the net forces and torques on the rotor in the x and y directions (the effect due to acceleration in the z-direction is small, and will later be dropped). Therefore, $[m_p, x_p, y_p, v_{xp}, v_{yp}, a_{xp}, a_{yp}]$ define the imbalance parameters for the PMI (the value of m_p is arbitrary).

Note that these *forces and torques are computed in the rotor frame*.

For example, F_x is the net force on the rotor aligned with the +x axis of the rotor. Since forces and torques aligned with the z-axis will not be measured by the strain gauges, they are not included here. The point-mass imbalance results in the following net forces and torques, with $F_{netIB_p} \triangleq [F_{xIB_p} \quad F_{yIB_p} \quad \tau_{xIB_p} \quad \tau_{yIB_p}]^T$:

F_g , the force due to gravity. This acts straight down at all times[***] and is independent of rotor motion. Its effect on the rotor appears as a torque equal to $m_p g r_p$, where r_p is the radial distance from the axis of rotation.

$$F_{netIB_{p,g}} = \begin{bmatrix} 0 & 0 & -m_p g y_p & m_p g x_p \end{bmatrix}^T \tag{36}$$

F_c , the centrifugal force. This acts radially outwards whenever the rotor is rotating, and is proportional to the angular velocity squared. Its effect on the rotor appears as a force in the direction of the point mass location vector (x_p, y_p) and with a magnitude equal to $m_p r_p \omega^2$.

$$F_{netIB_{p,c}} = \begin{bmatrix} m_p \omega^2 x_p & m_p \omega^2 y_p & 0 & 0 \end{bmatrix}^T \tag{37}$$

F_α , the force due to rotor angular acceleration. This force acts perpendicular to the point mass location vector (x_p, y_p) whenever the rotor has an angular acceleration, and is proportional to the angular acceleration. Its effect on the rotor is a force in the direction perpendicular to the point mass location vector and with a magnitude equal to $m_p r_p \alpha$.

$$F_{netIB_{p,\alpha}} = \begin{bmatrix} m_p \alpha y_p & -m_p \alpha x_p & 0 & 0 \end{bmatrix}^T \tag{38}$$

F_v , the Coriolis force due to the imbalance moving within the rotor. This force acts perpendicular to the imbalance velocity, (v_{xp}, v_{yp}) and is proportional to the imbalance velocity and the rotor angular velocity.

$$F_{netIB_{p,v}} = \begin{bmatrix} 2m_p \omega v_{yp} & -2m_p \omega v_{xp} & 0 & 0 \end{bmatrix}^T \tag{39}$$

F_a , the force due to the imbalance accelerating within the rotor. This acts in the opposite direction of the acceleration, (a_{xp}, a_{yp}, a_{zp}) .

$$F_{netIB_{p,a}} = \begin{bmatrix} -m_p a_{xp} & -m_p a_{yp} & -m_p a_{zp} y_p & m_p a_{zp} x_p \end{bmatrix}^T \tag{40}$$

2. Mass-couple imbalance

The second model imbalance, the mass-couple imbalance, consists of two point masses of mass m_c . The z-axis location of the upper (lower) point mass is chosen arbitrarily to lie in the same plane as the upper (lower) counterweights. Since the masses are located symmetrically about the geometric center of the rotor by definition, the x-y location of the upper point mass (x_c, y_c) also determines the lower point mass location $(-x_c, -y_c)$. There is no net force due to gravity, also due to symmetry. Positions (x_c, y_c) , velocities (v_{xc}, v_{yc}) , and accelerations (a_{xc}, a_{yc}) of the imbalance masses are always equal and opposite by definition. Therefore, $[m_c, x_c, y_c, v_{xc}, v_{yc}, a_{xc}, a_{yc}]$ define the imbalance parameters for the MCI. The MCI results in the following forces on the rotor in the rotor frame:

[***]Effects due to misalignment between the gravity vector and the axis of rotation are described in Section III.F

F_g, the force due to gravity, is zero by definition. This is because the masses are defined to be symmetrically located about the center of the rotor coordinate system

$$F_{netIB_{c,g}} = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{41}$$

F_c, the centrifugal force. This acts radially outwards whenever the rotor is rotating, and is proportional to the angular velocity squared.

$$F_{netIB_{c,c}} = \begin{bmatrix} 0 & 0 & -2m_c\omega^2 l y_c & 2m_c\omega^2 l x_c \end{bmatrix}^T \tag{42}$$

F_α, the force due to rotor angular acceleration. This force acts perpendicular to each point mass location vector whenever the rotor is accelerating, and is proportional to the angular acceleration.

$$F_{netIB_{c,\alpha}} = \begin{bmatrix} 0 & 0 & 2m_c\alpha l x_c & 2m_c\alpha l y_c \end{bmatrix}^T \tag{43}$$

F_v, the Coriolis force due to the imbalance moving within the rotor. This force acts perpendicular to the imbalance velocity, (v_{xc}, v_{yc}) and is proportional to the imbalance velocity and the rotor angular velocity.

$$F_{netIB_{c,v}} = \begin{bmatrix} 0 & 0 & 4m_c\omega l v_{xc} & 4m_c\omega l v_{yc} \end{bmatrix}^T \tag{44}$$

F_a, the force due to the imbalance accelerating within the rotor. This acts in the opposite direction of the acceleration, (a_{xc}, a_{yc}, a_{zc}).

$$F_{netIB_{c,a}} = \begin{bmatrix} 0 & 0 & 2m_c(a_{yc}l - a_{zc}y_c) & 2m_c(-a_{xc}l + a_{zc}x_c) \end{bmatrix}^T \tag{45}$$

Adding the net forces and torques on the rotor that are created by the two imbalances (forces are calculated in the rotor frame):

$$F_{netIB_p} = F_{netIB_{p,g}} + F_{netIB_{p,c}} + F_{netIB_{p,\alpha}} + F_{netIB_{p,v}} + F_{netIB_{p,a}}$$
$$F_{netIB_c} = F_{netIB_{c,g}} + F_{netIB_{c,c}} + F_{netIB_{c,\alpha}} + F_{netIB_{c,v}} + F_{netIB_{c,a}} \tag{46}$$
$$F_{netIB} = F_{netIB_p} + F_{netIB_c}$$

$$\begin{bmatrix} F_x \\ F_y \\ \tau_x \\ \tau_y \end{bmatrix}_{IB} = \begin{bmatrix} m_p\omega^2 x_p + m_p\alpha y_p + 2m_p\omega v_{yp} - m_p a_{xp} \\ m_p\omega^2 y_p - m_p\alpha x_p - 2m_p\omega v_{xp} - m_p a_{yp} \\ -m_p g y_p - m_p a_{zp} y_p \\ m_p g x_p + m_p a_{zp} x_p \end{bmatrix} + \cdots$$

$$\begin{bmatrix} 0 \\ 0 \\ -2m_c\omega^2 l y_c + 2m_c\alpha l x_c + 4m_c\omega l v_{xc} + 2m_c(a_{yc}l - a_{zc}y_c) \\ 2m_c\omega^2 l x_c + 2m_c\alpha l y_c + 4m_c\omega l v_{yc} + 2m_c(-a_{xc}l + a_{zc}x_c) \end{bmatrix} \tag{47}$$

$$
\begin{bmatrix} F_x \\ F_y \\ \tau_x \\ \tau_y \end{bmatrix}_{IB} =
\begin{bmatrix}
m_p \omega^2 x_p + m_p \alpha y_p + 2 m_p \omega v_{yp} - m_p a_{xp} \\
m_p \omega^2 y_p - m_p \alpha x_p - 2 m_p \omega v_{xp} - m_p a_{yp} \\
-m_p g y_p - m_p a_{zp} y_p - 2 m_c \omega^2 l y_c + 2 m_c \alpha l x_c + 4 m_c \omega l v_{xc} + 2 m_c (a_{yc} l - a_{zc} y_c) \\
m_p g x_p + m_p a_{zp} x_p + 2 m_c \omega^2 l x_c + 2 m_c \alpha l y_c + 4 m_c \omega l v_{yc} + 2 m_c (-a_{xc} l + a_{zc} x_c)
\end{bmatrix}
\tag{48}
$$

Putting the equations in a linear form so that the linear regression can be applied later requires dropping the second order terms (containing products of imbalance parameters). These are some of the terms due to acceleration of the imbalance within the rotor frame—likely to be small.

$$
\begin{bmatrix} F_x \\ F_y \\ \tau_x \\ \tau_y \end{bmatrix}_{IB} =
\begin{bmatrix}
m_p \omega^2 x_p + m_p \alpha y_p + 2 m_p \omega v_{yp} - m_p a_{xp} \\
m_p \omega^2 y_p - m_p \alpha x_p - 2 m_p \omega v_{xp} - m_p a_{yp} \\
-m_p g y_p - 2 m_c \omega^2 l y_c + 2 m_c \alpha l x_c + 4 m_c \omega l v_{xc} + 2 m_c l a_{yc} \\
m_p g x_p + 2 m_c \omega^2 l x_c + 2 m_c \alpha l y_c + 4 m_c \omega l v_{yc} - 2 m_c l a_{xc}
\end{bmatrix}
+
\begin{bmatrix}
0 \\
0 \\
-m_p a_{zp} y_p - 2 m_c a_{zc} y_c \\
m_p a_{zp} x_p + 2 m_c a_{zc} x_c
\end{bmatrix}
$$

$$
\approx
\begin{bmatrix}
m_p \omega^2 x_p + m_p \alpha y_p + 2 m_p \omega v_{yp} - m_p a_{xp} \\
m_p \omega^2 y_p - m_p \alpha x_p - 2 m_p \omega v_{xp} - m_p a_{yp} \\
-m_p g y_p - 2 m_c \omega^2 l y_c + 2 m_c \alpha l x_c + 4 m_c \omega l v_{xc} + 2 m_c l a_{yc} \\
m_p g x_p + 2 m_c \omega^2 l x_c + 2 m_c \alpha l y_c + 4 m_c \omega l v_{yc} - 2 m_c l a_{xc}
\end{bmatrix}
\tag{49}
$$

Putting these equations in matrix form, separating known/measured variables from the parameters to be identified,

$$
F_{netIB} = \Phi_{IB} \theta_{IB} = \Phi(m_p, m_c) \theta_{IB}
\tag{50}
$$

$$
\Phi_{IB} = \Phi(m_p, m_c) \triangleq
\begin{bmatrix}
m_p \omega^2 & m_p \alpha & 0 & 0 \\
-m_p \alpha & m_p \omega^2 & 0 & 0 \\
0 & -m_p g & 2 m_c \alpha l & -2 m_c \omega^2 l \\
m_p g & 0 & 2 m_c \omega^2 l & 2 m_c \alpha l
\end{bmatrix} \cdots
$$

$$
\begin{bmatrix}
0 & 2 m_p \omega & 0 & 0 & -m_p & 0 & 0 & 0 \\
-2 m_p \omega & 0 & 0 & 0 & 0 & -m_p & 0 & 0 \\
0 & 0 & 4 m_c \omega l & 0 & 0 & 0 & 0 & 2 m_c l \\
0 & 0 & 0 & 4 m_c \omega l & 0 & 0 & -2 m_c l & 0
\end{bmatrix}
\tag{51}
$$

$$
\theta_{IB} \triangleq \begin{bmatrix} x_p & y_p & x_c & y_c & v_{xp} & v_{yp} & v_{xc} & v_{yc} & a_{xp} & a_{yp} & a_{xc} & a_{yc} \end{bmatrix}^T
\tag{52}
$$

where θ_{IB} is a 12x1 column vector of the model-imbalance parameters. Φ is a 4x12 matrix that is a function of them point-mass and mass-couple mass magnitudes chosen. Done this way, Φ may be used later to model the counterweight forces and torques.

m_p, and m_c may be chosen arbitrarily, since they simply scale the values of the other parameters. In this analysis, they will be chosen to equal 1 (kilogram) so they will later drop out of the equations completely.

Note that the MCI produces torques only—no forces, and the PMI produces no torques except for the gravity effects.

While this equation contains the full effects of the imbalance on the net force and torque applied to the rotor, it does not follow that all terms should be included in the identification to follow. If the forces and torques due to imbalance velocity and acceleration (vs. position) are relatively small, and are also relatively difficult to identify

accurately due to the presence of noise and other disturbances, better identification of the imbalance position terms may be achieved by omitting the velocity and acceleration terms. The following equation shows the simplified case where both velocity and acceleration terms have been dropped.

$$F_{netIB} = \Phi_{IB}\theta_{IB}$$

$$\Phi_{IB} \triangleq \begin{bmatrix} m_p\omega^2 & m_p\alpha & 0 & 0 \\ -m_p\alpha & m_p\omega^2 & 0 & 0 \\ 0 & -m_pg & 2m_c\alpha l & -2m_c\omega^2 l \\ m_pg & 0 & 2m_c\omega^2 l & 2m_c\alpha l \end{bmatrix}, \theta_{IB} \triangleq \begin{bmatrix} x_p \\ y_p \\ x_c \\ y_c \end{bmatrix} \tag{53}$$

E. Net forces and torques due to the counterweights

If the forces and torques due to the counterweights can be calculated and subtracted from the measured forces and torques, the remaining signal can be used to identify the imbalance alone. The basic approach to deriving the effects of the counterweights is to calculate the equivalent set of model-imbalance parameters and then use the equations already derived in Section III.D.

Problem definition:

Using the same structure used to model the imbalance, counterweight parameters, m_{pCW}, m_{cCW}, and θ_{CW}, define the counterweight properties that result in net forces and torques on the rotor, where

$$\theta_{CW} \triangleq \begin{bmatrix} x_{pCW} & y_{pCW} & x_{cCW} & y_{cCW} & v_{xpCW} & v_{ypCW} & v_{xcCW} & v_{ycCW} \cdots \\ & & & a_{xpCW} & a_{ypCW} & a_{xcCW} & a_{ycCW} \end{bmatrix}^T \tag{54}$$

Unlike the model-imbalance parameters, which must be identified, the counterweight parameters can be calculated based on measurements of CW positions and/or motor command signals. These parameters can then be used, at each sample update, to calculate the net forces and torques applied to the rotor by the counterweights.

The counterweights are arranged in the layout shown in Figure 7. The positions of each of the four counterweights are described by $\begin{bmatrix} \delta_1 & \delta_2 & \delta_3 & \delta_4 \end{bmatrix}$. δ_1 is the counterweight in the upper plane in the +x direction, δ_2 is the counterweight in the upper plane in the +y direction, δ_3 is the counterweight in the lower plane in the +x direction, δ_4 is the counterweight in the lower plane, in the +y direction. m_{CW} is the mass of the moving portion of the counterweight assembly. The origins of the axes for δ_1, δ_2, δ_3, and δ_4 are chosen so that with $\delta = 0$, the net effect of the counterweight assembly is zero, regardless of where the counterweight actually is—this is the position where the moving counterweight is exactly counterbalanced by the corresponding fixed dead weight.

The velocities and accelerations are described by the first and second derivatives of the position signals, $\begin{bmatrix} \dot{\delta}_1 & \dot{\delta}_2 & \dot{\delta}_3 & \dot{\delta}_4 & \ddot{\delta}_1 & \ddot{\delta}_2 & \ddot{\delta}_3 & \ddot{\delta}_4 \end{bmatrix}$. Sensors will not be used to measure these signals directly (i.e., no accelerometer or tachometer feedback), and since the velocity and acceleration effects are likely to be small, they will not be calculated in the autobalancing system. However, their effects are included here and in the simulation for completeness. For this analysis, the counterweight coordinate vector, $\begin{bmatrix} \delta_1 & \delta_2 & \delta_3 & \delta_4 & \dot{\delta}_1 & \dot{\delta}_2 & \dot{\delta}_3 & \dot{\delta}_4 & \ddot{\delta}_1 & \ddot{\delta}_2 & \ddot{\delta}_3 & \ddot{\delta}_4 \end{bmatrix}$, is assumed to be available at all times.

The first step is to calculate the counterweight parameters, θ_{CW}, from the counterweight coordinate vector. The basic approach is to add the individual effects of each of the counterweights and select the counterweight parameters to yield the same effects. The equations are simplified using the fact that the counterweights all have mass equal to m_{CW} (this is the moving portion of the counterweights).
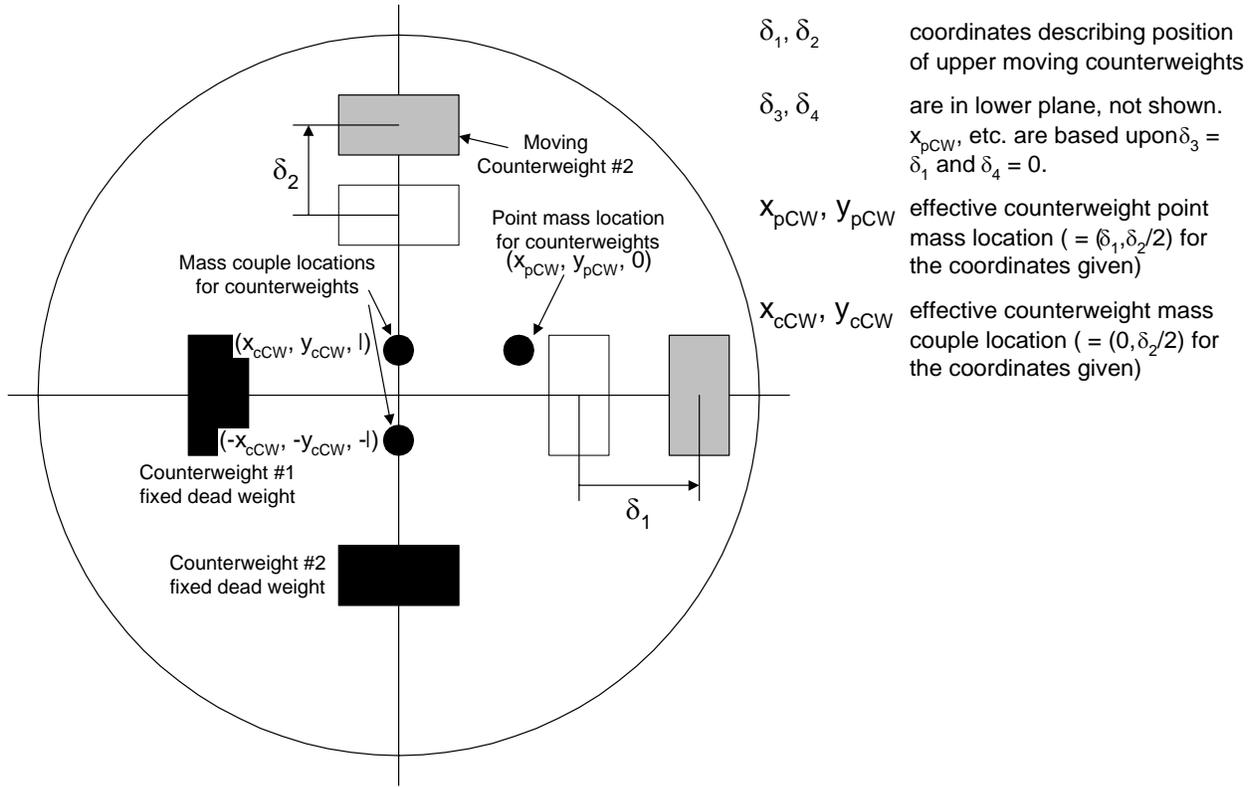
Figure 7. Counterweight coordinates

The figure labels include:

δ_1, δ_2 — coordinates describing position of upper moving counterweights

δ_3, δ_4 — are in lower plane, not shown. x_{pCW}, etc. are based upon $\delta_3 = \delta_1$ and $\delta_4 = 0$.

x_{pCW}, y_{pCW} — effective counterweight point mass location ($= (\delta_1, \delta_2/2)$ for the coordinates given)

x_{cCW}, y_{cCW} — effective counterweight mass couple location ($= (0, \delta_2/2)$ for the coordinates given)

Moving Counterweight #2

Point mass location for counterweights $(x_{pCW}, y_{pCW}, 0)$

Mass couple locations for counterweights

(x_{cCW}, y_{cCW}, l)

$(-x_{cCW}, -y_{cCW}, -l)$

Counterweight #1 fixed dead weight

Counterweight #2 fixed dead weight

$$
\begin{aligned}
m_{pCW} &= 4m_{CW} & m_{cCW} &= 2m_{CW} \\
x_{pCW} &= (\delta_1 + \delta_3)/4 & x_{cCW} &= (\delta_1 - \delta_3)/4 \\
y_{pCW} &= (\delta_2 + \delta_4)/4 & y_{cCW} &= (\delta_2 - \delta_4)/4 \\
v_{xpCW} &= \frac{d}{dt}x_{pCW} = (\dot{\delta}_1 + \dot{\delta}_3)/4 & v_{xcCW} &= \frac{d}{dt}x_{cCW} = (\dot{\delta}_1 - \dot{\delta}_3)/4 \\
v_{ypCW} &= \frac{d}{dt}y_{pCW} = (\dot{\delta}_2 + \dot{\delta}_4)/4 & v_{ycCW} &= \frac{d}{dt}y_{cCW} = (\dot{\delta}_2 - \dot{\delta}_4)/4 \\
a_{xpCW} &= \frac{d}{dt}v_{xpCW} = (\ddot{\delta}_1 + \ddot{\delta}_3)/4 & a_{xcCW} &= \frac{d}{dt}v_{xcCW} = (\ddot{\delta}_1 - \ddot{\delta}_3)/4 \\
a_{ypCW} &= \frac{d}{dt}v_{ypCW} = (\ddot{\delta}_2 + \ddot{\delta}_4)/4 & a_{ycCW} &= \frac{d}{dt}v_{ycCW} = (\ddot{\delta}_2 - \ddot{\delta}_4)/4
\end{aligned}
\tag{55}
$$

Since the counterweight parameters are defined the same as the imbalance parameters, a slightly modified version of Eq. (50) can be used to calculate the net CW force, without need to re-derive the physics. This means that the Φ from Eq. (51) can be used directly if m_{pCW} and m_{cCW} are used in place of m_p and m_c.

$$
F_{netCW} = \Phi_{CW}\theta_{CW} = \Phi(m_{pCW}, m_{cCW})\theta_{CW}
\tag{56}
$$

$$\Phi_{CW} = \Phi(m_{pCW}, m_{cCW}) \triangleq \begin{bmatrix} m_{pCW}\omega^2 & m_{pCW}\alpha & 0 & 0 \\ -m_{pCW}\alpha & m_{pCW}\omega^2 & 0 & 0 \\ 0 & -m_{pCW}g & 2m_{cCW}\alpha l & -2m_{cCW}\omega^2 l \\ m_{pCW}g & 0 & 2m_{cCW}\omega^2 l & 2m_{cCW}\alpha l \end{bmatrix} \cdots$$

$$\begin{bmatrix} 0 & 2m_{pCW}\omega & 0 & 0 & -m_{pCW} & 0 & 0 & 0 \\ -2m_{pCW}\omega & 0 & 0 & 0 & 0 & -m_{pCW} & 0 & 0 \\ 0 & 0 & 4m_{cCW}\omega l & 0 & 0 & 0 & 0 & 2m_{cCW}l \\ 0 & 0 & 0 & 4m_{cCW}\omega l & 0 & 0 & -2m_{cCW}l & 0 \end{bmatrix}$$

(57)

and θ_{CW} is defined in Eq. (54).

Depending on the accuracy of estimating the velocity and acceleration of the counterweights, it may be better to drop those terms from the autobalancing controller. The following equation shows the simplified case where both velocity and acceleration terms have been dropped.

$$F_{netCW} = \Phi_{CW}\theta_{CW}$$

$$\Phi_{CW} \triangleq \begin{bmatrix} m_{pCW}\omega^2 & m_{pCW}\alpha & 0 & 0 \\ -m_{pCW}\alpha & m_{pCW}\omega^2 & 0 & 0 \\ 0 & -m_{pCW}g & 2m_{cCW}\alpha l & -2m_{cCW}\omega^2 l \\ m_{pCW}g & 0 & 2m_{cCW}\omega^2 l & 2m_{cCW}\alpha l \end{bmatrix}, \theta_{CW} \triangleq \begin{bmatrix} x_{pCW} \\ y_{pCW} \\ x_{cCW} \\ y_{cCW} \end{bmatrix}$$

(58)

F. Net forces and torques due to axis misalignment during Earth-based testing

The misalignment between the rotor axis of rotation and the gravity vector will cause strain gauge measurements that are not due to the imbalances. If significant, these forces must be accounted for so that the model imbalance parameters may be properly identified. The misalignment is quantified by the misalignment angle in the x and y directions, ϕ_x and ϕ_y as shown in Figure 8. This misalignment causes a constant force on the fixed strain gauges and oscillating forces on the spinning strain gauges.

The misalignment causes forces due to the model imbalances and counterweights as well, but as these are second order effects, they will not be considered. These terms will not apply for the space station centrifuge, as it is in a micro-gravity environment.
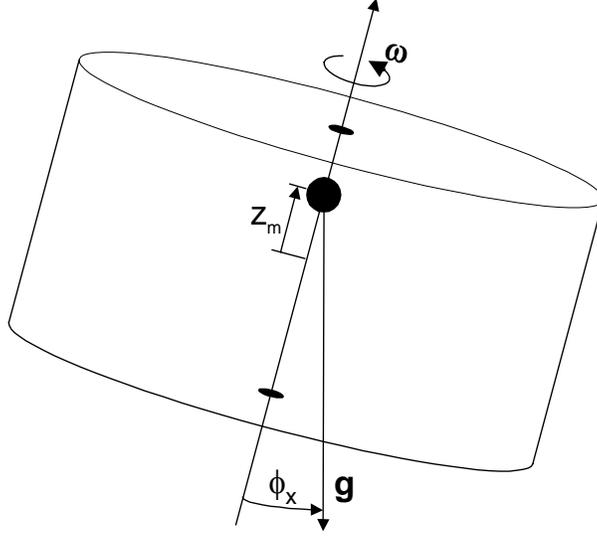
Figure 8. Misalignment between the axis of rotation and the gravity vector results in additional forces that are not due to imbalances (during Earth-based testing only)

$F_{netMisalignment}$, the force due to the axis misalignment, is affected by the total mass of the rotor, m, the z-axis location of the rotor center of mass (CM), z_m, and the misalignment angles, ϕ_x and ϕ_y. Angles are assumed to be small, so $\sin\phi \approx \phi$. It results in the following net forces and torques on the rotor, measured in the base frame (indicated by the "base" superscript in the following equations).

$$F_{netMisalignment}^{base} = \begin{bmatrix} mg\phi_x & mg\phi_y & -mg\phi_y z_m & mg\phi_x z_m \end{bmatrix}^T \tag{59}$$

Unlike the forces due to imbalances, these forces do not rotate with the rotor. To calculate the forces in the rotor frame, they are rotated by an angle $-\psi$:

$$F_{netMisalignment} = \begin{bmatrix} R(-\psi) & 0 \\ 0 & R(-\psi) \end{bmatrix} F_{netMisalignment}^{base} \tag{60}$$

$$F_{netMisalignment} = \begin{bmatrix} c\psi & s\psi & 0 & 0 \\ -s\psi & c\psi & 0 & 0 \\ 0 & 0 & c\psi & s\psi \\ 0 & 0 & -s\psi & c\psi \end{bmatrix} \begin{bmatrix} F_x^{base} \\ F_y^{base} \\ \tau_x^{base} \\ \tau_y^{base} \end{bmatrix} = \begin{bmatrix} mg\phi_x c\psi + mg\phi_y s\psi \\ -mg\phi_x s\psi + mg\phi_y c\psi \\ -mg\phi_y z_m c\psi + mg\phi_x z_m s\psi \\ mg\phi_y z_m s\psi + mg\phi_x z_m c\psi \end{bmatrix} \tag{61}$$

The parameters describing the imbalance must appear linearly in the equation if they are to be identified using a linear regression, so the torque terms containing $z_m\phi_x$ and $z_m\phi_y$ must be changed. New variables $z_{m\phi x} \triangleq z_m\phi_x$ and $z_{m\phi y} \triangleq z_m\phi_y$ are introduced to maintain linearity.

$$F_{netMisalignment} = \begin{bmatrix} mg\phi_x c\psi + mg\phi_y s\psi \\ -mg\phi_x s\psi + mg\phi_y c\psi \\ -mgz_{m\phi y} c\psi + mgz_{m\phi x} s\psi \\ mgz_{m\phi y} s\psi + mgz_{m\phi x} c\psi \end{bmatrix} \tag{62}$$

Putting these equations in matrix form, separating variables from the parameters to be identified,

$$F_{netMisalignment} = \Phi_{Misalignment} \theta_{Misalignment} \tag{63}$$

$$\Phi_{Misalignment} \triangleq mg \begin{bmatrix} c\psi & s\psi & 0 & 0 \\ -s\psi & c\psi & 0 & 0 \\ 0 & 0 & s\psi & -c\psi \\ 0 & 0 & c\psi & s\psi \end{bmatrix} \tag{64}$$

$$\theta_{Misalignment} \triangleq \begin{bmatrix} \phi_x \\ \phi_y \\ z_{m\phi x} \\ z_{m\phi y} \end{bmatrix} \tag{65}$$

$$F_{netMisalignment} = mg \begin{bmatrix} c\psi & s\psi & 0 & 0 \\ -s\psi & c\psi & 0 & 0 \\ 0 & 0 & s\psi & -c\psi \\ 0 & 0 & c\psi & s\psi \end{bmatrix} \begin{bmatrix} \phi_x \\ \phi_y \\ z_{m\phi x} \\ z_{m\phi y} \end{bmatrix} \tag{66}$$

Since $\theta_{Misalignment}$ will be identified (so that this effect may be subtracted out and not corrupt imbalance identification), mg may be chosen arbitrarily, so exact knowledge of the total rotor mass is not required. For example, if the chosen mg is too low, larger values for $\theta_{Misalignment}$ will result, but $F_{netMisalignment}$ will be the same. Using a fairly accurate value for mg will make the identified misalignment angles accurate, in case they are desired for some other reason.

G. Net force and torque noise

There are likely to be unmodeled net forces and torques on the rotor. Hopefully these will be small, random, and uncorrelated. If they are not, they should be identified separately (as was done for the axial misalignment effects). Here, such unknown net forces and torques are modeled as a white Gaussian noise source, E_{NFN}, where the "NFN" subscript indicates "Net Force Noise."

$$E_{NFN} = \begin{bmatrix} \varepsilon_{NFN1} & \varepsilon_{NFN2} & \varepsilon_{NFN3} & \varepsilon_{NFN4} \end{bmatrix}^T \tag{67}$$

H. Superposition of these forces

The linear matrix Eq. (68) is formed by adding the net forces and torques on the rotor, $F_{net} \triangleq [F_x \quad F_y \quad \tau_x \quad \tau_y]^T$, due to the model-imbalance parameters, θ_{IB}, counterweight parameters, θ_{CW}, and axis misalignment parameters, $\theta_{Misalignment}$,

$$\begin{aligned} F_{net} &= F_{netIB} + F_{netCW} + F_{netMisalignment} + E_{NFN} \\ &= \Phi(m_p, m_c)\theta_{IB} + \Phi(m_{pCW}, m_{cCW})\theta_{CW} + \Phi_{Misalignment}\theta_{Misalignment} + E_{NFN} \end{aligned} \tag{68}$$

where each term is specified in an earlier equation:

$\Phi(m_p, m_c)$ – Equation (51) – calculable using known parameters and measurements

θ_{IB} – Equation (52) – unknown, will be identified on-line

$\Phi(m_{pCW}, m_{cCW})$ – Equation (57) – calculable using known parameters and measurements

θ_{CW} – Equation (54) – calculable using known parameters and measurements

$\Phi_{Misalignment}$ – Equation (64) – calculable using known parameters and measurements

$\theta_{Misalignment}$ – Equation (65) – unknown, may be identified on-line or calibrated off-line

E_{NFN} – Equation (67) – noise

Section III.A presented a linear regression method to estimate the net force, $\hat{F}_{net}$, at each sample period using the available strain gauges. Note that that method operates completely independently of any consideration of imbalances, counterweights, etc.

The following sections are aimed at manipulating Eq. (68) to enable identification of θ_{IB} and $\theta_{Misalignment}$ so that $\hat{\theta}_{IB}$ may be used to guide the counterweight positioning.

I. Identification of the unknown parameters

The net forces and torques on the rotor are due to the sum of the imbalance, counterweight, and axial misalignment forces as shown in Eq. (68), repeated here.

$$F_{net} = F_{netIB} + F_{netCW} + F_{netMisalignment} + E_{NFN}$$
$$= \Phi(m_p, m_c)\theta_{IB} + \Phi(m_{pCW}, m_{cCW})\theta_{CW} + \Phi_{Misalignment}\theta_{Misalignment} + E_{NFN}$$

(68)

$\hat{F}_{net}$ can be estimated based upon the strain gauge measurements, as shown in Eq. (32). F_{netCW} can be calculated based upon the counterweight coordinates, as shown in Eq. (56) or (58).

The basic approach of the steps to follow is to manipulate Eq. (68) into the "regression form" shown in Eq. (1) and then to solve for the unknown parameters. The choice of parameters to be included in this identification is a design issue that needs to be optimized through experimentation with simulation and actual data. This section derives the regression-form equations for selected groups of variables to be identified.

Depending on the noise level present and the motion characteristics of the imbalance, it may not be practical to identify the imbalance velocity or acceleration parameters (elements 5-8 and 9-12, respectively in θ_{IB}). It may be that better results can be achieved by removing these parameters from the identification and treating their effects as noise.

All terms are used in the simulation, whether or not they are identified.

Regardless of how many parameters are to be identified, only the first 4 elements of θ_{IB} (imbalance position) are used for autobalancing. These 4 elements define the position of the PMI and MCI. Identifying extra parameters, if possible, is important since they prevent these useful parameters from being corrupted by the extra forces that they represent.

1. Identifying imbalance position (only)

In this approach, the simplest of the options presented, only $\begin{bmatrix} x_p & y_p & x_c & y_c \end{bmatrix}$ are identified. It is assumed that:

- Strain gauge biases are available from some prior calibration or other method independent of this identification.

- The axis misalignment parameters are available from some prior calibration or other method independent of this identification.

- The counterweight parameters, θ_{CW}, are known

- The effects of the imbalance velocity and acceleration are sufficiently small or difficult to identify that better results are obtained by ignoring them, and treating them as unmodeled disturbances (noise).

Substituting $\hat{F}_{net}$ from Eq. (32) for F_{net} in Eq. (68), and using $\hat{B}$, $\hat{\theta}_{CW}$, and $\hat{\theta}_{Misalignment}$ to indicate the parameters that are considered known.

$$\Gamma T^{-1}(S - \hat{B}) = \Phi(m_p, m_c)\theta_{IB} + \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} + \Phi_{Misalignment}\hat{\theta}_{Misalignment} + E_{NFN} \qquad (69)$$

$$\Phi(m_p, m_c)\theta_{IB} = \Gamma T^{-1}(S - \hat{B}) - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} - \Phi_{Misalignment}\hat{\theta}_{Misalignment} - E_{NFN} \qquad (70)$$

Separating θ_{IB} into parts that will be identified vs. ignored (Φ_{IB} is also similarly partitioned),

$$\theta_{IB} \triangleq \begin{bmatrix} \theta_{IB_{pos}} & \theta_{IB_{vel+acc}} \end{bmatrix}^T \qquad (71)$$

$$\theta_{IB_{pos}} \triangleq \begin{bmatrix} x_p & y_p & x_c & y_c \end{bmatrix}^T \qquad (72)$$

$$\theta_{IB_{vel+acc}} \triangleq \begin{bmatrix} v_{xp} & v_{yp} & v_{xc} & v_{yc} & a_{xp} & a_{yp} & a_{xc} & a_{yc} \end{bmatrix}^T \qquad (73)$$

$$\Phi_{IB}\theta_{IB} = \Phi_{IB_{pos}}\theta_{IB_{pos}} + \Phi_{IB_{vel+acc}}\theta_{IB_{vel+acc}} =$$
$$\Gamma T^{-1}(S - \hat{B}) - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} - \Phi_{Misalignment}\hat{\theta}_{Misalignment} - E_{NFN} \qquad (74)$$

$$\Phi_{IB_{pos}}\theta_{IB_{pos}} = \Gamma T^{-1}(S - \hat{B}) - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} - \Phi_{Misalignment}\hat{\theta}_{Misalignment} \cdots$$
$$-\Phi_{IB_{vel+acc}}\theta_{IB_{vel+acc}} - E_{NFN} \qquad (75)$$

As long as the assumption that $\Phi_{IB_{vel+acc}}\theta_{IB_{vel+acc}}$ can be treated as noise is true, all terms on the right (except the noise) are known parameters or measurements, and the unknown parameters on the left side appear linearly. So this is now in a form enabling least squares solution of the $\theta_{IB_{pos}}$ vector using the standard RLS algorithm presented earlier. The equation is now in regression form, $Ax = b + \varepsilon$, where

$$A = \Phi_{IB_{pos}}$$
$$x = \theta_{IB_{pos}}$$
$$b = \Gamma T^{-1}(S - \hat{B}) - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} - \Phi_{Misalignment}\hat{\theta}_{Misalignment} \qquad (76)$$
$$\varepsilon = -\Phi_{IB_{vel+acc}}\theta_{IB_{vel+acc}} - E_{NFN}$$

It may be the case that axis misalignment forces are negligible, in which case that term can be omitted.

Since the goal is to provide an imbalance estimate in real time, so that it can be corrected with counterweights, a recursive least squares (RLS) implementation is used, as presented in Ref. 8 and summarized in Appendix B. Since the goal is to track a moving imbalance, an exponentially weighted RLS is chosen, with the effective time window chosen according to the expected rate of imbalance change and desired control bandwidth.

2. *Identifying imbalance position and axial misalignment*

In this approach, $\begin{bmatrix} x_p & y_p & x_c & y_c & \phi_x & \phi_y & z_{m\phi x} & z_{m\phi y} \end{bmatrix}$ are identified. It is assumed that:

- Strain gauge biases are available from some prior calibration or other method independent of this identification.

- The counterweight parameters, θ_{CW}, are known

- The effects of the imbalance velocity and acceleration are sufficiently small or difficult to identify that better results are obtained by ignoring them, and treating them as unmodeled disturbances (noise).

Again substituting $\hat{F}_{net}$ from Eq. (32) for F_{net} in Eq. (68), and using $\hat{B}$, $\hat{\theta}_{CW}$, and $\hat{\theta}_{Misalignment}$ to indicate the parameters that are considered known.

$$\Gamma T^{-1}(S - \hat{B}) = \Phi(m_p, m_c)\theta_{IB} + \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} + \Phi_{Misalignment}\theta_{Misalignment} + E_{NFN} \tag{77}$$

$$\Phi_{IB_{pos}}\theta_{IB_{pos}} + \Phi_{IB_{vel+acc}}\theta_{IB_{vel+acc}} + \Phi_{Misalignment}\theta_{Misalignment} =$$
$$\Gamma T^{-1}(S - \hat{B}) - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} - E_{NFN} \tag{78}$$

$$\Phi_{IB_{pos}}\theta_{IB_{pos}} + \Phi_{Misalignment}\theta_{Misalignment} =$$
$$\Gamma T^{-1}(S - \hat{B}) - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} - \Phi_{IB_{vel+acc}}\theta_{IB_{vel+acc}} - E_{NFN} \tag{79}$$

The unknown parameters, $\theta_{IB_{pos}}$ and $\theta_{Misalignment}$, in Eq. (79) will now be identified simultaneously, yielding the imbalance position and axial misalignment parameters. The misalignment parameters are not used for any purpose, but if their effect was not included in the identification, the imbalance estimation would be biased. This identification of imbalance parameters is only relevant for Earth-based testing, where gravity is significant. If $\theta_{Misalignment}$ is found to be fairly constant, it may be better to identify that in an off-line process, which would reduce the degrees of freedom to be identified on-line, probably improving accuracy.

Concatenating the Φ matrices and θ vectors,

$$\Phi_{IB_{pos}+Misalignment} \triangleq \begin{bmatrix} \Phi_{pos} & \Phi_{Misalignment} \end{bmatrix} \tag{80}$$

$$\theta_{IB_{pos}+Misalignment} \triangleq \begin{bmatrix} \theta_{IB_{pos}} \\ \theta_{Misalignment} \end{bmatrix} \tag{81}$$

Equation (79) may be re-written as

$$\Phi_{IB_{pos}+Misalignment}\theta_{IB_{pos}+Misalignment} =$$
$$\Gamma T^{-1}(S - \hat{B}) - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} - \Phi_{IB_{vel+acc}}\theta_{IB_{vel+acc}} - E_{NFN} \tag{82}$$

which is in regression form with

$$\begin{aligned}
A &= \Phi_{IB_{pos}+Misalignment} \\
x &= \theta_{IB_{pos}+Misalignment} \\
b &= \Gamma T^{-1}(S - \hat{B}) - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} \\
\varepsilon &= -\Phi_{IB_{vel+acc}}\theta_{IB_{vel+acc}} - E_{NFN}
\end{aligned} \tag{83}$$

An unfortunate issue for ground-based experiments is that $\theta_{IB_{pos}}$ and $\theta_{Misalignment}$ must be updated with the same RLS time constant, whereas it is likely that the imbalance requires faster tracking than the axial misalignment – so if the two sets of parameters are identified simultaneously, the $\theta_{Misalignment}$ estimates will not be as accurate as they could be if the RLS time constant were increased.

3. Comparison with Kalman Filter

This highlights a difference between the least squares approach presented and the possibility of using a Kalman Filter (KF) for identification. With the present LS approach, there is no connection between the identified positions, velocities, and accelerations. For example, the position estimate may be constantly changing, but the velocity and acceleration estimates may be zero (not likely, but possible). However, in a KF approach, the positions, velocities, and accelerations are all coupled (for example, velocity would be an integration of acceleration, etc.), and the

statistical properties of imbalance motion could be specified. The KF filter would be optimized based on these statistical properties, whereas the RLS approach requires tuning of the exponential weighting time constant to achieve this result. Also, in a KF approach, the fact that misalignment parameters will vary much more slowly than imbalance parameters can be set directly, and the identification result will reflect this.

The LS approach was chosen to avoid the added complexity of the KF, which is a concern both from a computational processing issue and reduced algorithmic robustness issue. But the KF design and implementation was not performed, so it should be considered an option for further exploration.

4. Identifying imbalance, axial misalignment, and strain gauge biases

The previous sections showed how imbalance (and axial misalignment) parameters can be identified, if the strain gauge biases are known through some prior calibration. This information is reflected in the calculation of $\hat{F}_{net}$ in Eq. (32). This section derives the equations to identify the biases on-line simultaneously with the imbalance parameter ID. Depending on the bias drift rate, it may actually be better to identify the biases with a separate process that assumes the average imbalance force is zero (as it would be if the autobalancing system were operating correctly). Alternatively, if the bias drift rate is very slow, it may be possible to identify them with an off-line process.

The derivation in this section also includes identification of velocity and acceleration imbalance parameters, and is the most complex option presented.

In this approach, the following 24-element vector is identified, which includes all 12 imbalance parameters, the 4 axial misalignment parameters, and the 8 strain gauge biases:

$$\theta_{IB+Misalignment+B} \triangleq \begin{bmatrix} \theta_{IB} \\ \theta_{Misalignment} \\ B \end{bmatrix} \tag{84}$$

where

$$\theta_{IB} \triangleq \begin{bmatrix} x_p & y_p & x_c & y_c & v_{xp} & v_{yp} & v_{xc} & v_{yc} & a_{xp} & a_{yp} & a_{xc} & a_{yc} \end{bmatrix}^T \tag{52}$$

$$\theta_{Misalignment} \triangleq \begin{bmatrix} \phi_x & \phi_y & z_{m\phi x} & z_{m\phi y} \end{bmatrix}^T \tag{65}$$

$$B \triangleq \begin{bmatrix} \beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \beta_6 & \beta_7 & \beta_8 \end{bmatrix}^T \tag{8}$$

It is assumed that the counterweight parameters, θ_{CW}, are known.

Again substituting $\hat{F}_{net}$ from Eq. (32) for F_{net} in Eq. (68), and using $\hat{\theta}_{CW}$ to indicate the parameters that are considered known.

$$\Gamma T^{-1}(S-B) = \Phi(m_p, m_c)\theta_{IB} + \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} + \Phi_{Misalignment}\theta_{Misalignment} + E_{NFN} \tag{85}$$

$$\Phi(m_p, m_c)\theta_{IB} + \Phi_{Misalignment}\theta_{Misalignment} + \Gamma T^{-1}B = \Gamma T^{-1}S - \Phi(m_{pCW}, m_{cCW})\theta_{CW} - E_{NFN} \tag{86}$$

The unknown parameters, θ_{IB}, $\theta_{Misalignment}$, and B in Eq. (86) will now be identified simultaneously, yielding the imbalance, axial misalignment, and strain gauge bias parameters. Concatenating the Φ matrices and θ vectors,

$$\Phi_{IB+Misalignment+B} \triangleq \begin{bmatrix} \Phi(m_p, m_c) & \Phi_{Misalignment} & \Gamma T^{-1} \end{bmatrix} \tag{87}$$

Equation (86) may now be written as

$$\Phi_{IB+Misalignment+B}\theta_{IB+Misalignment+B} = \Gamma T^{-1}S - \Phi(m_{pCW}, m_{cCW})\theta_{CW} - E_{NFN} \tag{88}$$

which is in regression form with

$$A = \Phi_{IB+Misalignment+B}$$

$$x = \theta_{IB+Misalignment+B}$$

$$b = \Gamma T^{-1} S - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} \tag{89}$$

$$\varepsilon = -E_{NFN}$$

The ΓT^{-1} term in the preceding equations is a 4x8 matrix that can be calculated analytically to save run-time computation as follows.

$$\Gamma T^{-1} = \begin{bmatrix} \Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\ 0 & \Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} \\ 0 & \Gamma_{32} & 0 & \Gamma_{34} & 0 & \Gamma_{36} & 0 & \Gamma_{38} \\ -\Gamma_{32} & 0 & -\Gamma_{34} & 0 & -\Gamma_{36} & 0 & -\Gamma_{38} & 0 \end{bmatrix} \cdots$$

$$\begin{bmatrix} c\psi & s\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ -s\psi & c\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c\psi & s\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & -s\psi & c\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{90}$$

$$\Gamma T^{-1} = \begin{bmatrix} \Gamma_{11}c\psi & \Gamma_{11}s\psi & \Gamma_{13}c\psi & \Gamma_{13}s\psi & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\ -\Gamma_{11}s\psi & \Gamma_{11}c\psi & -\Gamma_{13}s\psi & \Gamma_{13}c\psi & 0 & \Gamma_{15} & 0 & \Gamma_{17} \\ -\Gamma_{32}s\psi & \Gamma_{32}c\psi & -\Gamma_{34}s\psi & \Gamma_{34}c\psi & 0 & \Gamma_{36} & 0 & \Gamma_{38} \\ -\Gamma_{32}c\psi & -\Gamma_{32}s\psi & -\Gamma_{34}c\psi & -\Gamma_{34}s\psi & -\Gamma_{36} & 0 & -\Gamma_{38} & 0 \end{bmatrix} \tag{91}$$

J. Summary of imbalance identification algorithms

As mentioned earlier, there are some options regarding whether all 12 imbalance parameters are to be identified, whether axial misalignment is to be identified or calibrated beforehand, and whether the strain gauge biases are to be identified or calibrated beforehand. The equations corresponding to some of the permutations on these decisions are summarized here.

One goal of this section is to repeat enough of the equations so that all the steps in implementing the algorithm can be seen, without concern for showing the complete derivation.

1. Identification of 12 imbalance parameters, axial misalignment, and strain gauge biases

This is the most complex case, using the regression equation derived in Section III.I.4 as Eq. (88), repeated here. Equations specifying these terms are repeated below.

$$\Phi_{IB+Misalignment+B}\theta_{IB+Misalignment+B} = \Gamma T^{-1} S - \Phi(m_{pCW}, m_{cCW})\theta_{CW} - E_{NFN} \tag{88}$$

$$\Phi_{IB+Misalignment+B} \triangleq \begin{bmatrix} \Phi(m_p, m_c) & \Phi_{Misalignment} & \Gamma T^{-1} \end{bmatrix} \tag{87}$$

$$\Phi(m_p, m_c) \triangleq \begin{bmatrix} m_p\omega^2 & m_p\alpha & 0 & 0 \\ -m_p\alpha & m_p\omega^2 & 0 & 0 \\ 0 & -m_p g & 2m_c\alpha l & -2m_c\omega^2 l \\ m_p g & 0 & 2m_c\omega^2 l & 2m_c\alpha l \end{bmatrix} \cdots$$

$$\begin{matrix} 0 & 2m_p\omega & 0 & 0 & -m_p & 0 & 0 & 0 \\ -2m_p\omega & 0 & 0 & 0 & 0 & -m_p & 0 & 0 \\ 0 & 0 & 4m_c\omega l & 0 & 0 & 0 & 0 & 2m_c l \\ 0 & 0 & 0 & 4m_c\omega l & 0 & 0 & -2m_c l & 0 \end{matrix}$$

(51)

$$\Phi_{Misalignment} \triangleq mg \begin{bmatrix} c\psi & s\psi & 0 & 0 \\ -s\psi & c\psi & 0 & 0 \\ 0 & 0 & s\psi & -c\psi \\ 0 & 0 & c\psi & s\psi \end{bmatrix}$$

(64)

$$c\psi \triangleq \cos\psi$$
$$s\psi \triangleq \sin\psi$$

(19)

$$\Gamma T^{-1} = \begin{bmatrix} \Gamma_{11}c\psi & \Gamma_{11}s\psi & \Gamma_{13}c\psi & \Gamma_{13}s\psi & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\ -\Gamma_{11}s\psi & \Gamma_{11}c\psi & -\Gamma_{13}s\psi & \Gamma_{13}c\psi & 0 & \Gamma_{15} & 0 & \Gamma_{17} \\ -\Gamma_{32}s\psi & \Gamma_{32}c\psi & -\Gamma_{34}s\psi & \Gamma_{34}c\psi & 0 & \Gamma_{36} & 0 & \Gamma_{38} \\ -\Gamma_{32}c\psi & -\Gamma_{32}s\psi & -\Gamma_{34}c\psi & -\Gamma_{34}s\psi & -\Gamma_{36} & 0 & -\Gamma_{38} & 0 \end{bmatrix}$$

(91)

ΓT^{-1} can be calculated analytically, as shown here, to save computation time. The Γ_{ij} are constant functions of the strain gauge geometry, so only need to be calculated once.

$$\Gamma \triangleq (G^T G)^{-1} G^T = \begin{bmatrix} \Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\ 0 & \Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} \\ 0 & \Gamma_{32} & 0 & \Gamma_{34} & 0 & \Gamma_{36} & 0 & \Gamma_{38} \\ -\Gamma_{32} & 0 & -\Gamma_{34} & 0 & -\Gamma_{36} & 0 & -\Gamma_{38} & 0 \end{bmatrix}$$

(34)

$$G = \begin{bmatrix} \dfrac{-z_{Bf}}{z_{Af} - z_{Bf}} & 0 & 0 & \dfrac{1}{z_{Af} - z_{Bf}} \\[4mm] 0 & \dfrac{-z_{Bf}}{z_{Af} - z_{Bf}} & \dfrac{-1}{z_{Af} - z_{Bf}} & 0 \\[4mm] \dfrac{z_{Af}}{z_{Af} - z_{Bf}} & 0 & 0 & \dfrac{-1}{z_{Af} - z_{Bf}} \\[4mm] 0 & \dfrac{z_{Af}}{z_{Af} - z_{Bf}} & \dfrac{1}{z_{Af} - z_{Bf}} & 0 \\[4mm] \dfrac{-z_{Bs}}{z_{As} - z_{Bs}} & 0 & 0 & \dfrac{1}{z_{As} - z_{Bs}} \\[4mm] 0 & \dfrac{-z_{Bs}}{z_{As} - z_{Bs}} & \dfrac{-1}{z_{As} - z_{Bs}} & 0 \\[4mm] \dfrac{z_{As}}{z_{As} - z_{Bs}} & 0 & 0 & \dfrac{-1}{z_{As} - z_{Bs}} \\[4mm] 0 & \dfrac{z_{As}}{z_{As} - z_{Bs}} & \dfrac{1}{z_{As} - z_{Bs}} & 0 \end{bmatrix} \qquad (17)$$

So $\Phi_{IB+Misalignment+B}$ is a function of the following constant known parameters: $m_p, m_c, g, l, z_{Af}, z_{Bf}, z_{As}, z_{Bs}$ and time-varying variables which can be measured or estimated: ψ, ω, α. Since m_p, m_c are arbitrary, they may be set to 1 kg to reduce computations.

$$\theta_{IB+Misalignment+B} \triangleq \begin{bmatrix} \theta_{IB} \\ \theta_{Misalignment} \\ B \end{bmatrix} \qquad (84)$$

$$\theta_{IB} \triangleq \begin{bmatrix} x_p & y_p & x_c & y_c & v_{xp} & v_{yp} & v_{xc} & v_{yc} & a_{xp} & a_{yp} & a_{xc} & a_{yc} \end{bmatrix}^T \qquad (52)$$

$$\theta_{Misalignment} \triangleq \begin{bmatrix} \phi_x \\ \phi_y \\ z_{m\phi x} \\ z_{m\phi y} \end{bmatrix} \qquad (65)$$

$$B = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \\ \beta_6 \\ \beta_7 \\ \beta_8 \end{bmatrix} \qquad (8)$$

So the parameters to be identified consist of 4 positions, 4 velocities, 4 accelerations, 4 axial misalignment parameters, and 8 strain gauge biases.

Continuing with the right side of the regression equation,

$$S = \begin{bmatrix} S_{Axf} \\ S_{Ayf} \\ S_{Bxf} \\ S_{Byf} \\ S_{Axs} \\ S_{Ays} \\ S_{Bxs} \\ S_{Bys} \end{bmatrix} \qquad (8)$$

$$\Phi(m_{pCW}, m_{cCW}) \triangleq \begin{bmatrix} m_{pCW}\omega^2 & m_{pCW}\alpha & 0 & 0 \\ -m_{pCW}\alpha & m_{pCW}\omega^2 & 0 & 0 \\ 0 & -m_{pCW}g & 2m_{cCW}\alpha l & -2m_{cCW}\omega^2 l \\ m_{pCW}g & 0 & 2m_{cCW}\omega^2 l & 2m_{cCW}\alpha l \end{bmatrix} \cdots$$

$$\begin{matrix} 0 & 2m_{pCW}\omega & 0 & 0 & -m_{pCW} & 0 & 0 & 0 \\ -2m_{pCW}\omega & 0 & 0 & 0 & 0 & -m_{pCW} & 0 & 0 \\ 0 & 0 & 4m_{cCW}\omega l & 0 & 0 & 0 & 0 & 2m_{cCW}l \\ 0 & 0 & 0 & 4m_{cCW}\omega l & 0 & 0 & -2m_{cCW}l & 0 \end{matrix} \qquad (57)$$

$$\theta_{CW} \triangleq \begin{bmatrix} x_{pCW} & y_{pCW} & x_{cCW} & y_{cCW} & v_{xpCW} & v_{ypCW} & v_{xcCW} & v_{ycCW} \cdots \\ & & & a_{xpCW} & a_{ypCW} & a_{xcCW} & a_{ycCW} \end{bmatrix}^T \qquad (54)$$

$$m_{pCW} = 4m_{CW} \qquad\qquad m_{cCW} = 2m_{CW}$$

$$x_{pCW} = (\delta_1 + \delta_3)/4 \qquad\qquad x_{cCW} = (\delta_1 - \delta_3)/4$$

$$y_{pCW} = (\delta_2 + \delta_4)/4 \qquad\qquad y_{cCW} = (\delta_2 - \delta_4)/4$$

$$v_{xpCW} = \frac{d}{dt} x_{pCW} = (\dot\delta_1 + \dot\delta_3)/4 \qquad v_{xcCW} = \frac{d}{dt} x_{cCW} = (\dot\delta_1 - \dot\delta_3)/4 \qquad (55)$$

$$v_{ypCW} = \frac{d}{dt} y_{pCW} = (\dot\delta_2 + \dot\delta_4)/4 \qquad v_{ycCW} = \frac{d}{dt} y_{cCW} = (\dot\delta_2 - \dot\delta_4)/4$$

$$a_{xpCW} = \frac{d}{dt} v_{xpCW} = (\ddot\delta_1 + \ddot\delta_3)/4 \qquad a_{xcCW} = \frac{d}{dt} v_{xcCW} = (\ddot\delta_1 - \ddot\delta_3)/4$$

$$a_{ypCW} = \frac{d}{dt} v_{ypCW} = (\ddot\delta_2 + \ddot\delta_4)/4 \qquad a_{ycCW} = \frac{d}{dt} v_{ycCW} = (\ddot\delta_2 - \ddot\delta_4)/4$$

So the right side of Eq. (88) is a function of the following constant known parameters: $m_{CW}, g, l, z_{Af}, z_{Bf}, z_{As}, z_{Bs}$ and time-varying variables which can be measured or estimated: $\psi, \omega, \alpha, S_{Axf}, S_{Ayf}, S_{Bxf}, S_{Byf}, S_{Axs}, S_{Ays}, S_{Bxs}, S_{Bys}, \delta_1, \delta_2, \delta_3, \delta_4, \dot\delta_1, \dot\delta_2, \dot\delta_3, \dot\delta_4, \ddot\delta_1, \ddot\delta_2, \ddot\delta_3, \ddot\delta_4$. The first 3 variables are the rotor position, velocity, and acceleration. The next 8 are the 4 fixed and 4 spinning strain gauge measurements. The final 12 are the position, velocity, and acceleration of the counterweights.

To implement this algorithm
- Upon initialization,
    - Calculate Γ
    - Initialize the parameter estimation error covariance matrix, P_0, referred to in Appendix B, and used by the RLS algorithm. Ref. 8 contains some suggestions for this. Setting it as a diagonal matrix with large values on the diagonal is probably not a bad guess – this implies no prior knowledge.
    - Initialize $\theta_{IB+Misalignment+B}$ with the best prior estimates. This may be a vector of zeros.
- At each update
    - Sample the rotor encoder, tachometer (if present), strain gauges, and CW encoders
    - Use the rotor encoder, tachometer (if present), and spin motor current (if available) to estimate ψ, ω, α
    - Filter and combine the individual strain gauges as described in Section III.A
    - Use the CW encoder measurements and commands (if available) to estimate the position, velocity, and acceleration of the counterweights.
    - Calculate $\Phi_{IB+Misalignment+B}$ using the known parameters and variables as summarized above. This will be a 4x24 matrix, represented by ϕ_k^T in the RLS algorithm in Appendix B.
    - Calculate $\Gamma T^{-1} S - \Phi(m_{pCW}, m_{cCW})\theta_{CW}$. This will be a 4x1 vector, represented by y in the RLS algorithm in Appendix B.
    - "Modification 2" from Appendix B applies here, so update $\hat\theta_{IB+Misalignment+B}$ as described there.
    - Use the first four elements of $\hat\theta_{IB+Misalignment+B}$ to drive the counterweights, as shown in Eq. (120), or in the case of counterweight failure, Eq. (122).

2. Identification of 4 imbalance parameters only

This is the simplest case, using the regression equation derived in Section III.I.1 as Eq. (75), repeated here. Equations specifying these terms are repeated below. This is the simplest case. It assumes axial misalignment parameters and strain gauge biases are known, based on prior calibration. The imbalance velocities and accelerations are judged to be small enough that the identification degrees of freedom they provide do more harm than good.

$$\Phi_{IB_{pos}} \theta_{IB_{pos}} = \Gamma T^{-1}(S - \hat{B}) - \Phi(m_{pCW}, m_{cCW})\hat{\theta}_{CW} - \Phi_{Misalignment}\hat{\theta}_{Misalignment} \cdots$$

$$- \Phi_{IB_{vel+acc}} \theta_{IB_{vel+acc}} - E_{NFN}$$

(75)

$$\Phi_{IB_{pos}} \triangleq \begin{bmatrix} m_p \omega^2 & m_p \alpha & 0 & 0 \\ -m_p \alpha & m_p \omega^2 & 0 & 0 \\ 0 & -m_p g & 2m_c \alpha l & -2m_c \omega^2 l \\ m_p g & 0 & 2m_c \omega^2 l & 2m_c \alpha l \end{bmatrix}$$

(92)

$$\theta_{IB_{pos}} \triangleq \begin{bmatrix} x_p & y_p & x_c & y_c \end{bmatrix}^T$$

(72)

So $\Phi_{IB_{pos}}$ is a function of the following constant known parameters: m_p, m_c, g, l and time-varying variables which can be measured or estimated: ω, α. Since m_p, m_c are arbitrary, they may be set to 1 kg to reduce computations.

Continuing with the right side of the regression equation,

$$\Gamma T^{-1} = \begin{bmatrix} \Gamma_{11} c\psi & \Gamma_{11} s\psi & \Gamma_{13} c\psi & \Gamma_{13} s\psi & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\ -\Gamma_{11} s\psi & \Gamma_{11} c\psi & -\Gamma_{13} s\psi & \Gamma_{13} c\psi & 0 & \Gamma_{15} & 0 & \Gamma_{17} \\ -\Gamma_{32} s\psi & \Gamma_{32} c\psi & -\Gamma_{34} s\psi & \Gamma_{34} c\psi & 0 & \Gamma_{36} & 0 & \Gamma_{38} \\ -\Gamma_{32} c\psi & -\Gamma_{32} s\psi & -\Gamma_{34} c\psi & -\Gamma_{34} s\psi & -\Gamma_{36} & 0 & -\Gamma_{38} & 0 \end{bmatrix}$$

(91)

ΓT^{-1} can be calculated analytically, as shown here, to save computation time. The Γ_{ij} are constant functions of the strain gauge geometry, so only need to be calculated once.

$$\Gamma \triangleq (G^T G)^{-1} G^T = \begin{bmatrix} \Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\ 0 & \Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} \\ 0 & \Gamma_{32} & 0 & \Gamma_{34} & 0 & \Gamma_{36} & 0 & \Gamma_{38} \\ -\Gamma_{32} & 0 & -\Gamma_{34} & 0 & -\Gamma_{36} & 0 & -\Gamma_{38} & 0 \end{bmatrix}$$

(34)

American Institute of Aeronautics and Astronautics

$$G = \begin{bmatrix} \dfrac{-z_{Bf}}{z_{Af} - z_{Bf}} & 0 & 0 & \dfrac{1}{z_{Af} - z_{Bf}} \\[3mm] 0 & \dfrac{-z_{Bf}}{z_{Af} - z_{Bf}} & \dfrac{-1}{z_{Af} - z_{Bf}} & 0 \\[3mm] \dfrac{z_{Af}}{z_{Af} - z_{Bf}} & 0 & 0 & \dfrac{-1}{z_{Af} - z_{Bf}} \\[3mm] 0 & \dfrac{z_{Af}}{z_{Af} - z_{Bf}} & \dfrac{1}{z_{Af} - z_{Bf}} & 0 \\[3mm] \dfrac{-z_{Bs}}{z_{As} - z_{Bs}} & 0 & 0 & \dfrac{1}{z_{As} - z_{Bs}} \\[3mm] 0 & \dfrac{-z_{Bs}}{z_{As} - z_{Bs}} & \dfrac{-1}{z_{As} - z_{Bs}} & 0 \\[3mm] \dfrac{z_{As}}{z_{As} - z_{Bs}} & 0 & 0 & \dfrac{-1}{z_{As} - z_{Bs}} \\[3mm] 0 & \dfrac{z_{As}}{z_{As} - z_{Bs}} & \dfrac{1}{z_{As} - z_{Bs}} & 0 \end{bmatrix} \qquad (17)$$

$$\text{c}\,\psi \triangleq \cos\psi$$
$$\text{s}\,\psi \triangleq \sin\psi \qquad (19)$$

$$S = \begin{bmatrix} S_{Axf} \\ S_{Ayf} \\ S_{Bxf} \\ S_{Byf} \\ S_{Axs} \\ S_{Ays} \\ S_{Bxs} \\ S_{Bys} \end{bmatrix} \qquad (8)$$

$$\hat{B} = \begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \\ \hat{\beta}_3 \\ \hat{\beta}_4 \\ \hat{\beta}_5 \\ \hat{\beta}_6 \\ \hat{\beta}_7 \\ \hat{\beta}_8 \end{bmatrix} \qquad (93)$$

$$\Phi(m_{pCW}, m_{cCW}) \triangleq \begin{bmatrix} m_{pCW}\omega^2 & m_{pCW}\alpha & 0 & 0 \\ -m_{pCW}\alpha & m_{pCW}\omega^2 & 0 & 0 \\ 0 & -m_{pCW}g & 2m_{cCW}\alpha l & -2m_{cCW}\omega^2 l \\ m_{pCW}g & 0 & 2m_{cCW}\omega^2 l & 2m_{cCW}\alpha l \end{bmatrix} \cdots$$

$$\begin{bmatrix} 0 & 2m_{pCW}\omega & 0 & 0 & -m_{pCW} & 0 & 0 & 0 \\ -2m_{pCW}\omega & 0 & 0 & 0 & 0 & -m_{pCW} & 0 & 0 \\ 0 & 0 & 4m_{cCW}\omega l & 0 & 0 & 0 & 0 & 2m_{cCW}l \\ 0 & 0 & 0 & 4m_{cCW}\omega l & 0 & 0 & -2m_{cCW}l & 0 \end{bmatrix} \qquad (57)$$

$$\theta_{CW} \triangleq \begin{bmatrix} x_{pCW} & y_{pCW} & x_{cCW} & y_{cCW} & v_{xpCW} & v_{ypCW} & v_{xcCW} & v_{ycCW} \cdots \\ & & & & a_{xpCW} & a_{ypCW} & a_{xcCW} & a_{ycCW} \end{bmatrix}^T \qquad (54)$$

$$m_{pCW} = 4m_{CW} \qquad\qquad m_{cCW} = 2m_{CW}$$
$$x_{pCW} = (\delta_1 + \delta_3)/4 \qquad\qquad x_{cCW} = (\delta_1 - \delta_3)/4$$
$$y_{pCW} = (\delta_2 + \delta_4)/4 \qquad\qquad y_{cCW} = (\delta_2 - \delta_4)/4$$
$$v_{xpCW} = \frac{d}{dt}x_{pCW} = (\dot{\delta}_1 + \dot{\delta}_3)\Big/4 \qquad v_{xcCW} = \frac{d}{dt}x_{cCW} = (\dot{\delta}_1 - \dot{\delta}_3)\Big/4 \qquad (55)$$
$$v_{ypCW} = \frac{d}{dt}y_{pCW} = (\dot{\delta}_2 + \dot{\delta}_4)\Big/4 \qquad v_{ycCW} = \frac{d}{dt}y_{cCW} = (\dot{\delta}_2 - \dot{\delta}_4)\Big/4$$
$$a_{xpCW} = \frac{d}{dt}v_{xpCW} = (\ddot{\delta}_1 + \ddot{\delta}_3)\Big/4 \qquad a_{xcCW} = \frac{d}{dt}v_{xcCW} = (\ddot{\delta}_1 - \ddot{\delta}_3)\Big/4$$
$$a_{ypCW} = \frac{d}{dt}v_{ypCW} = (\ddot{\delta}_2 + \ddot{\delta}_4)\Big/4 \qquad a_{ycCW} = \frac{d}{dt}v_{ycCW} = (\ddot{\delta}_2 - \ddot{\delta}_4)\Big/4$$

$$\Phi_{Misalignment} \triangleq mg \begin{bmatrix} c\psi & s\psi & 0 & 0 \\ -s\psi & c\psi & 0 & 0 \\ 0 & 0 & s\psi & -c\psi \\ 0 & 0 & c\psi & s\psi \end{bmatrix} \qquad (64)$$

$$\hat{\theta}_{Misalignment} \triangleq \begin{bmatrix} \hat{\phi}_x \\ \hat{\phi}_y \\ \hat{z}_{m\phi x} \\ \hat{z}_{m\phi y} \end{bmatrix} \tag{65}$$

K. Summary of noise sources modeled

E_{NFN} is the "Net-Force noise" signal, indicating a true net force and torque on the rotor.

E_{SFN} is the "Strain gauge Force Noise" signal, indicating forces at the strain gauges that did not result in net forces.

E_{SSN} is the "Strain gauge Sensor Noise" signal, indicating sensor errors not resulting from actual forces.

L. SG fault detection isolation and reconfiguration (FDIR)

The FDI approach taken here is based upon maximum likelihood theory. It is loosely analogous to the commonly applied bank of Kalman filters. A key advancement over an earlier attempt using this approach was to simplify the model to include only geometric relationships–so that imbalance ID results, dynamic modeling, etc. are not needed. This is very important for simplicity and robustness.

The method for estimating $\hat{F}_{net}$, derived in Section III.B and resulting in the following repeated equation,

$$\hat{F}_{net} = \left(\left(G^T G \right)^{-1} G^T \right) \left(T^{-1} \left(S - B \right) \right), \tag{31}$$

is repeated once for each possible failure mode. For the problem statement studied and implemented in simulation, 9 different conditions are considered: (1) the case of no failures, and then (2-9) once for each situation where a single strain gauge would have failed. The failure modes are modeled so that when a SG fails it reads zero plus noise. This is not really the case here, since it is assumed that each SG is in a pair reading the exact opposite force, but this assumption makes the FDI problem very challenging. For actual hardware implementation, the actual failure modes would be properly calculated.

For the case where S_i has failed, the estimation of $\hat{F}_{net}$ is re-derived here as follows. Recalling the governing equation,

$$S - B = TGF_{net} + \left(E_{SFN} + E_{SSN} \right) \tag{25}$$

If S_i has failed so it reads zero plus noise, this equation can be re-written as

$$S - B = I_i TGF_{net} + \left(E_{SFN} + E_{SSN} \right) \tag{94}$$

where I_i is a modified identity matrix, with ones on the diagonal, except $I_i(i,i) = 0$. So S_i will read zero plus noise, as desired.[†††] Unfortunately, $I_i T$ is not invertible, so the equation cannot be pre-multiplied by $(I_i T)^{-1}$ as was done before for computational efficiency. Note that $I_i T$ is simply T with the ith row replaced with zeros. This equation is now in regression form where

---

[†††] Another failure mode might be that the SG signal could read anything. In that case, a better model might be to remove the equation for that SG from the system of equations. That could be achieved by removing the ith row from the T matrix and the S and B vectors, giving them each 7 rows.

$$A = I_i T G$$
$$x = F_{net}$$
$$b = S - B$$
$$\varepsilon = -(E_{SFN} + E_{SSN})$$
(95)

So the least-squares solution for this equation is

$$\hat{F}_{net_i} = \left((I_i T G)^T (I_i T G) \right)^{-1} (I_i T G)^T (S - B).$$
(96)

Unlike the solution for no failures, the $(A^T A)^{-1} A^T$ term is a function of T, which is a function of rotor angle, ψ, so it must be updated at each sample.

Once $\hat{F}_{net_i}$ is calculated for each failure mode, i, (including the case of no failures), at each sample update, k, the residual, $r_{i,k}$, from the least squares calculation is found as follows.

$$r_{i,k} \triangleq \left\| (S - B)_{measured,k} - (S - B)_{estimated,assuming\ i\ failed,k} \right\|$$
(97)

$$r_{i,k} = \left\| (S - B)_{measured,k} - I_i T G \hat{F}_{net_{i,k}} \right\|$$
(98)

$$r_{i,k} = \left\| (S - B)_{measured,k} - I_i T_k G \left((I_i T_k G)^T (I_i T_k G) \right)^{-1} (I_i T_k G)^T (S - B)_{measured,k} \right\|$$
(99)

As derived below, the one of the 9 candidate models that most closely matches the actual measurements (calculated as the minimum residual, averaged over a window of time) is selected as the correct one, providing detection and isolation of the failure mode.

The uncertainty model for the maximum likelihood calculation is simplified as follows[9]. Since the uncertainty in SG measurements comes from multiple sources, as summarized earlier, for simplicity it is modeled as coming from a single combined source for the maximum likelihood tests. The residual in the preceding equations is considered to represent the single combined uncertainty. The variance of this (assumed Gaussian and un-correlated) distribution is calculated in simulation, for the case of no SG failures.

At each sample time, k, the residual, $r_{i^*,k}$, for the true failure mode, i^*, is modeled as a continuous random variable having the following normal probability density function

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$
(100)

Making the assumption that the residual will have zero mean for the true failure mode,

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}},$$
(101)

where σ^2 is the variance calculated in simulation, as mentioned above.

For N measurements at discrete times $1, 2, \cdots, N$, the joint probability distribution function is

$$F(x_1, x_2, \cdots, x_N) = \Pr(X_1 < x_1, X_2 < x_2, \cdots, X_N < x_N)$$
$$= \int_{-\infty}^{x_1} \int_{-\infty}^{x_2} \cdots \int_{-\infty}^{x_N} f(x_1, x_2, \cdots, x_N) dx_1 dx_2 \cdots dx_N,$$
(102)

where $f(x_1, x_2, \cdots, x_N)$ is the joint probability density function. Since the measurements are independent and identically distributed (IID),

$$F(x_1, x_2, \cdots, x_N) = \Pr(X_1 < x_1)\Pr(X_2 < x_2)\Pr(\cdots)\Pr(X_N < x_N) \tag{103}$$

$$F(x_1, x_2, \cdots, x_N) = \prod_{k=1}^{N} \Pr(X_k < x_k). \tag{104}$$

The joint probability density function is

$$f(x_1, x_2, \cdots, x_N) = \prod_{k=1}^{N} f(x_k). \tag{105}$$

Substituting the individual probability density function from Eq. (101)

$$\begin{aligned}
f(x_1, x_2, \cdots, x_N) &= \prod_{k=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x_k^2}{2\sigma^2}} \\
&= \left(2\pi\sigma^2\right)^{-\frac{N}{2}} \prod_{k=1}^{N} e^{-\frac{x_k^2}{2\sigma^2}} \\
&= \left(2\pi\sigma^2\right)^{-\frac{N}{2}} e^{-\frac{1}{2\sigma^2}\sum_{k=1}^{N} x_k^2}
\end{aligned} \tag{106}$$

The likelihood function corresponding to the 9 candidate cases, from $i = 0\ldots8$, is

$$L(i) = f(x_1, x_2, \cdots, x_N; i). \tag{107}$$

The maximum likelihood failure mode is chosen as the i that maximizes this likelihood function, or, more conveniently (and equivalently), $\ln L(i)$

$$\begin{aligned}
\ln L(i) &= \ln\left(f(x_1, x_2, \cdots, x_N; i)\right) \\
&= \ln\left(\left(2\pi\sigma^2\right)^{-\frac{N}{2}} e^{-\frac{1}{2\sigma^2}\sum_{k=1}^{N} x_{i,k}^2}\right) \\
&= \ln\left(\left(2\pi\sigma^2\right)^{-\frac{N}{2}}\right) + \ln\left(e^{-\frac{1}{2\sigma^2}\sum_{k=1}^{N} x_{i,k}^2}\right) \\
&= -\frac{N}{2}\ln\left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2}\sum_{k=1}^{N} x_{i,k}^2
\end{aligned} \tag{108}$$

With σ^2 and N known, this expression for $\ln L(i)$ is maximized by minimizing the summation. The maximum likelihood failure mode is chosen with the following expression,

$$i^* = \min_i \sum_{k=1}^{N} x_{i,k}^2, \tag{109}$$

where $x_{i,k}$ is the residual calculated at time k, using $(S - B)_{estimated, assuming\ i\ failed, k}$ (for example, $x_{i,k} = r_{i,k}$ from Eq. (98)). N determines the size of the window used (the number of most recent samples).

A generalized likelihood ratio test is performed between the best and second best fit to prevent detection when signals and residuals are small (as is true in a balanced condition). The ratio is taken between the summations, as shown in Eq. (109). In simulation testing, this approach has proven sufficiently robust to detect intermittent and changing SG faults without requiring significant additional logic.

In summary, the SG FDIR algorithm does the following during each pass through the sample loop:

1. Sample spin angle encoder and estimate ψ. Use this to calculate T and T^{-1}.

2. Sample strain gauges and calculate the $(S - B)$ vector.

3. Calculate $\hat{F}_{net}$ for the case of no SG failures present, indicated as $\hat{F}_{net_0}$, using Eq. (32).

4. Calculate $\hat{F}_{net}$ for each SG failure case to be considered, indicated as $\hat{F}_{net_1} \cdots \hat{F}_{net_8}$, using Eq. (96).

5. Calculate the residuals associated with each potential fault mode, $r_{0,k} \cdots r_{8,k}$, using Eq. (98).

6. Compute the running modified likelihood function for each potential fault mode, $\sum r_0^2, \cdots, \sum r_8^2$, as the sum over the most recent N samples (where N is tuned based on the desired rate of response in FDI vs. the noise level present).

7. Find the closest and second closest fault mode matches, $i_{closest}$ and $i_{second\,closest}$, corresponding to the lowest and second-to-lowest values of $\sum r_i^2$. Compute the generalized likelihood ratio as the ratio of these values, $\gamma = \dfrac{\sum r_{i_{closest}}^2}{\sum r_{i_{second\,closest}}^2}$.

8. If γ is below a specified threshold, and $i_{closest}$ is different than the presently isolated fault mode, $i_{isolated}$, declare $i_{closest}$ as the isolated fault mode (set $i_{isolated} = i_{closest}$). This step accomplishes the fault detection and isolation (FDI). Here, the detection and isolation are performed at the same step, which is not always the case.

9. Use $\hat{F}_{net_{i\,isolated}}$ (already calculated in step 4, above) as $\hat{F}_{net}$ in the remaining autobalancing calculations (e.g., use $\hat{F}_{net_{i\,isolated}}$ where $\Gamma T^{-1}(S - \hat{B})$ is used). This step implements the reconfiguration (R), if needed.

An important result of the approach taken is that the calculations used for FDI (i.e., the calculation of $\hat{F}_{net_i}$) can then be used directly in the reconfiguration (R). Reconfiguration involves simply using the $\hat{F}_{net_i}$ value corresponding to the failure mode isolated, $\hat{F}_{net_{i\,closest}}$, rather than the $\hat{F}_{net}$ for the case of no failures, $\hat{F}_{net_0}$.

Another key feature of this approach is that it is independent of the dynamical model and disturbance forces that may be acting on the rotor. It simply looks for self-consistency among the various redundant sensors that measure forces on the (assumed rigid) rotor. This approach can be applied directly to the ISS Centrifuge design, even though that has a compliant suspension and uses displacement sensors instead of force sensors.

IV. Counterweight control and FDIR

Counterweight control refers to the portion of the overall control system that takes in estimates of imbalance parameters and drives the CWs to null the total imbalance. The actual low level control system (e.g., servo loop or stepper motor drive system) to move the counterweights is not critical (as long as it has sufficient bandwidth) and is not discussed here. So this section refers to generating the commanded CW positions that are to be sent to the lower level control systems.

The 12 parameters defining the position, velocity, and acceleration of the PMI and MCI are updated at each sample, based upon the calculations summarized in Section III.J.

$$\theta_{IB} \triangleq \begin{bmatrix} x_p & y_p & x_c & y_c & v_{xp} & v_{yp} & v_{xc} & v_{yc} & a_{xp} & a_{yp} & a_{xc} & a_{yc} \end{bmatrix}^T \tag{52}$$

Assuming that imbalance motions (if any) are basically random and likely to be faster than the control bandwidth of the counterweights, the goal here will be to move the counterweights to counteract the imbalance positions,

$$\theta_{IB_{pos}} \triangleq \begin{bmatrix} x_p & y_p & x_c & y_c \end{bmatrix}^T$$

For the counterweights to statically negate the imbalance,

$$\begin{bmatrix} m_p & 0 & 0 & 0 \\ 0 & m_p & 0 & 0 \\ 0 & 0 & m_c & 0 \\ 0 & 0 & 0 & m_c \end{bmatrix} \hat{\theta}_{IB_{pos}} + \begin{bmatrix} m_{pCW} & 0 & 0 & 0 \\ 0 & m_{pCW} & 0 & 0 \\ 0 & 0 & m_{cCW} & 0 \\ 0 & 0 & 0 & m_{cCW} \end{bmatrix} \theta_{CW_{pos,desired}} = 0 . \tag{110}$$

Solving for $\theta_{CW_{pos,desired}}$,

$$\theta_{CW_{pos,desired}} = - \begin{bmatrix} m_{pCW}^{-1} & 0 & 0 & 0 \\ 0 & m_{pCW}^{-1} & 0 & 0 \\ 0 & 0 & m_{cCW}^{-1} & 0 \\ 0 & 0 & 0 & m_{cCW}^{-1} \end{bmatrix} \begin{bmatrix} m_p & 0 & 0 & 0 \\ 0 & m_p & 0 & 0 \\ 0 & 0 & m_c & 0 \\ 0 & 0 & 0 & m_c \end{bmatrix} \hat{\theta}_{IB_{pos}} . \tag{111}$$

Recalling the equations defining the counterweight (position) parameters,

$$\begin{aligned} m_{pCW} &= 4m_{CW} & m_{cCW} &= 2m_{CW} \\ x_{pCW} &= (\delta_1 + \delta_3)/4 & x_{cCW} &= (\delta_1 - \delta_3)/4 . \\ y_{pCW} &= (\delta_2 + \delta_4)/4 & y_{cCW} &= (\delta_2 - \delta_4)/4 \end{aligned} \tag{55}$$

In matrix form,

$$\theta_{CW_{pos}} = m_{CW} \begin{bmatrix} m_{pCW}^{-1} & 0 & 0 & 0 \\ 0 & m_{pCW}^{-1} & 0 & 0 \\ 0 & 0 & 0.5m_{cCW}^{-1} & 0 \\ 0 & 0 & 0 & 0.5m_{cCW}^{-1} \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} . \tag{112}$$

Solving for δ in terms of $\theta_{CW_{pos}}$

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} = m_{CW}^{-1} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}^{-1} \begin{bmatrix} m_{pCW}^{-1} & 0 & 0 & 0 \\ 0 & m_{pCW}^{-1} & 0 & 0 \\ 0 & 0 & 0.5m_{cCW}^{-1} & 0 \\ 0 & 0 & 0 & 0.5m_{cCW}^{-1} \end{bmatrix}^{-1} \theta_{CW_{pos}} \tag{113}$$

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} = \frac{1}{2m_{CW}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} m_{pCW} & 0 & 0 & 0 \\ 0 & m_{pCW} & 0 & 0 \\ 0 & 0 & 2m_{cCW} & 0 \\ 0 & 0 & 0 & 2m_{cCW} \end{bmatrix} \theta_{CW_{pos}} . \tag{114}$$

Substituting $\theta_{CW_{pos,desired}}$ from Eq. (111) for $\theta_{CW_{pos}}$ in Eq. (114),

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} = \frac{-1}{2m_{CW}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} m_{pCW} & 0 & 0 & 0 \\ 0 & m_{pCW} & 0 & 0 \\ 0 & 0 & 2m_{cCW} & 0 \\ 0 & 0 & 0 & 2m_{cCW} \end{bmatrix} \cdots$$
$$\begin{bmatrix} m_{pCW}^{-1} & 0 & 0 & 0 \\ 0 & m_{pCW}^{-1} & 0 & 0 \\ 0 & 0 & m_{cCW}^{-1} & 0 \\ 0 & 0 & 0 & m_{cCW}^{-1} \end{bmatrix} \begin{bmatrix} m_p & 0 & 0 & 0 \\ 0 & m_p & 0 & 0 \\ 0 & 0 & m_c & 0 \\ 0 & 0 & 0 & m_c \end{bmatrix} \hat{\theta}_{IB_{pos}} \tag{115}$$

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} = \frac{-1}{2m_{CW}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} m_p & 0 & 0 & 0 \\ 0 & m_p & 0 & 0 \\ 0 & 0 & 2m_c & 0 \\ 0 & 0 & 0 & 2m_c \end{bmatrix} \hat{\theta}_{IB_{pos}} \tag{116}$$

$$\begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} = \frac{-1}{2m_{CW}} \begin{bmatrix} m_p \hat{x}_p + 2m_c \hat{x}_c \\ m_p \hat{y}_p + 2m_c \hat{y}_c \\ m_p \hat{x}_p - 2m_c \hat{x}_c \\ m_p \hat{y}_p - 2m_c \hat{y}_c \end{bmatrix}. \tag{117}$$

These are the desired counterweight coordinates to be sent to the servo control loop.

Recall that m_p and m_c are arbitrary, and could be set to 1 kg to simplify the equations. Whichever values are chosen will directly scale $\hat{\theta}_{IB}$. This mapping between $\hat{\theta}_{IB}$ and δ is determined exactly because the CW control policy chosen has 4 degrees of freedom. If a different counterweight configuration, perhaps having more than 4 degrees of freedom were chosen, there would be some added flexibility in this mapping.

A. Counterweight fault detection and isolation

Since encoders measure the location of individual CWs for both the ISS and SSRL Centrifuges, CW FDI is relatively straightforward. The algorithm simply looks for a mismatch between the commanded and sensed positions. An enduring mismatch, with an unchanging sensed position indicates a failure, either of the CW or its encoder. The simple approach taken now is to consider that CW frozen, and stop trying to drive it. If needed, it should be possible to determine whether the CW encoder or drive system has failed, but it would be relatively difficult, and provide limited value considering that CW redundancy exists.

B. Redundantly driven counterweights

So far, the analysis follows the counterweight configuration shown in Figure 7, in which only one of the CWs in a redundant pair moves, and the other remains fixed. One simple alternative to this would be to drive both counterweights, where δ indicates the total motion of each CW in the redundant pairs. The net δ could then be calculated as

$$\begin{aligned} \delta_1 &= \delta_{1A} + \delta_{1B} \\ \delta_2 &= \delta_{2A} + \delta_{2B} \\ \delta_3 &= \delta_{3A} + \delta_{3B} \\ \delta_4 &= \delta_{4A} + \delta_{4B} \end{aligned}, \tag{118}$$

where, for example, δ_{1A} is the position shown as δ_1 in Figure 7, and δ_{1B} is the position of the CW labeled "counterweight #1 fixed dead weight", which is positive in the same direction as δ_{1A}. Eq. (118) can be substituted into the preceding equations, including Eq. (117), as follows

$$
\begin{bmatrix} \delta_{1A} + \delta_{1B} \\ \delta_{2A} + \delta_{2B} \\ \delta_{3A} + \delta_{3B} \\ \delta_{4A} + \delta_{4B} \end{bmatrix} = \frac{-1}{2m_{CW}} \begin{bmatrix} m_p \hat{x}_p + 2m_c \hat{x}_c \\ m_p \hat{y}_p + 2m_c \hat{y}_c \\ m_p \hat{x}_p - 2m_c \hat{x}_c \\ m_p \hat{y}_p - 2m_c \hat{y}_c \end{bmatrix}.
\tag{119}
$$

This relation is used for standard CW control, in which the A and B CWs in each pair are nominally driven to have the same coordinates (i.e., $\delta_{1A_{command}} = \delta_{1B_{command}}$, etc.). So in this case, the desired δ is calculated using Eq. (117), then individual CW desired positions are calculated as follows

$$
\begin{bmatrix} \delta_{1A} \\ \delta_{2A} \\ \delta_{1B} \\ \delta_{2B} \\ \delta_{3A} \\ \delta_{3B} \\ \delta_{4A} \\ \delta_{4B} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} = \frac{-1}{4m_{CW}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m_p \hat{x}_p + 2m_c \hat{x}_c \\ m_p \hat{y}_p + 2m_c \hat{y}_c \\ m_p \hat{x}_p - 2m_c \hat{x}_c \\ m_p \hat{y}_p - 2m_c \hat{y}_c \end{bmatrix}.
\tag{120}
$$

C. Reconfiguration for counterweight failures

If a CW fails by freezing in place, the other one in the pair will be commanded to make up the difference. With the CW geometry used here, the fault reconfiguration is fairly obvious. For example, CW 1A freezes at position δ_{1A}^{frozen}, the commanded CW vector becomes

$$
\begin{bmatrix} \delta_{1A} \\ \delta_{2A} \\ \delta_{1B} \\ \delta_{2B} \\ \delta_{3A} \\ \delta_{3B} \\ \delta_{4A} \\ \delta_{4B} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \\ \delta_4 \end{bmatrix} + \begin{bmatrix} \delta_{1A}^{frozen} \\ 0 \\ -\delta_{1A}^{frozen} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
\tag{121}
$$

$$
\begin{bmatrix} \delta_{1A} \\ \delta_{2A} \\ \delta_{1B} \\ \delta_{2B} \\ \delta_{3A} \\ \delta_{3B} \\ \delta_{4A} \\ \delta_{4B} \end{bmatrix} = \frac{-1}{4m_{CW}} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m_p \hat{x}_p + 2m_c \hat{x}_c \\ m_p \hat{y}_p + 2m_c \hat{y}_c \\ m_p \hat{x}_p - 2m_c \hat{x}_c \\ m_p \hat{y}_p - 2m_c \hat{y}_c \end{bmatrix} + \begin{bmatrix} \delta_{1A}^{frozen} \\ 0 \\ -\delta_{1A}^{frozen} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{122}
$$

For other geometries, a more complex mapping may be required, but as long as there is physical redundancy for all degrees of freedom, such a mapping will exist.

V. MATLAB simulation

Basic results confirm that noise should not be a significant problem. One-Newton noise levels in the strain gauges can be tolerated easily. Gravity imbalance forces dominate. These are affected by the distance between the strain gauge clusters, as well as the rotation speed. Also, a higher speed will reduce the effect of sensor noise, as the centrifugal forces are increased by the square of the angular rate.

The recursive least squares algorithm runs at 3 kHz on a 2.2 GHz Pentium 4 in MATLAB[10], which includes simulation as well as the autobalancing calculations. This will slow down if more equations are added (more strain gauges), digital filtering is required, or a slower processor is used. However, the code could be sped up if programmed in C.

Sample screen outputs from the graphical user interface shown in the following figures, which were taken from a simulation in which the centrifuge was generating 1.0 g (setpoint), the counterweight motion was not enabled (resulting in much higher imbalance forces), and strain gauge pair #4 (non-rotating, measuring y-axis force in the lower plane) has failed.
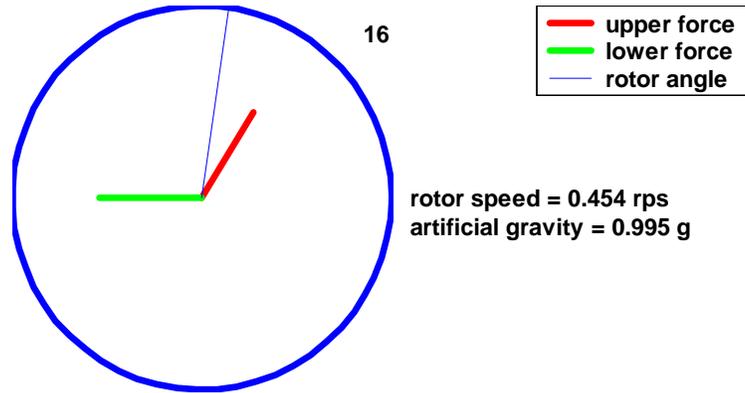

Figure 9. Software simulation GUI – bearing forces

This view shows the measured bearing forces, including sensor noise, vibrations, and imbalances. The blue line indicates the zero-degree angle of the rotor. The red and green lines are vectors indicating the measured forces in the upper and lower planes. Since SG #4 has failed, the green vector has very little y-component (only noise). The scale factor (16) indicates that the radius of the blue circle is 16 Newtons, very large because the counterweight motion has been disabled in this simulation.
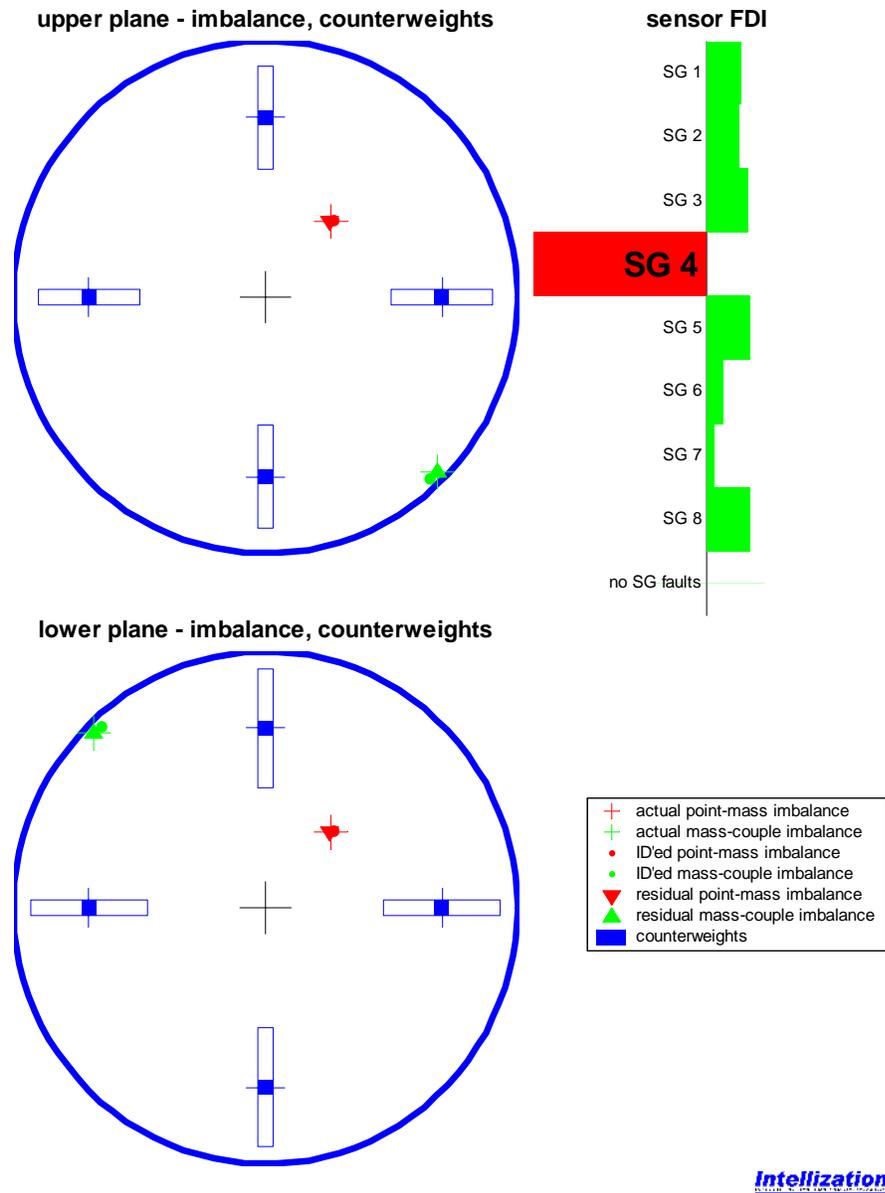
Figure 10. **Software simulation GUI – imbalance tracking, Strain gauge FDIR**

The state of imbalance, imbalance-identification, and imbalance-cancellation are shown in the left figure for upper and lower counterweight planes. The results of the strain gauge FDI are shown in the upper right, indicating correctly that SG #4 has failed. Automatic reconfiguration following the fault isolation enables the identification to remain very accurate, as shown.
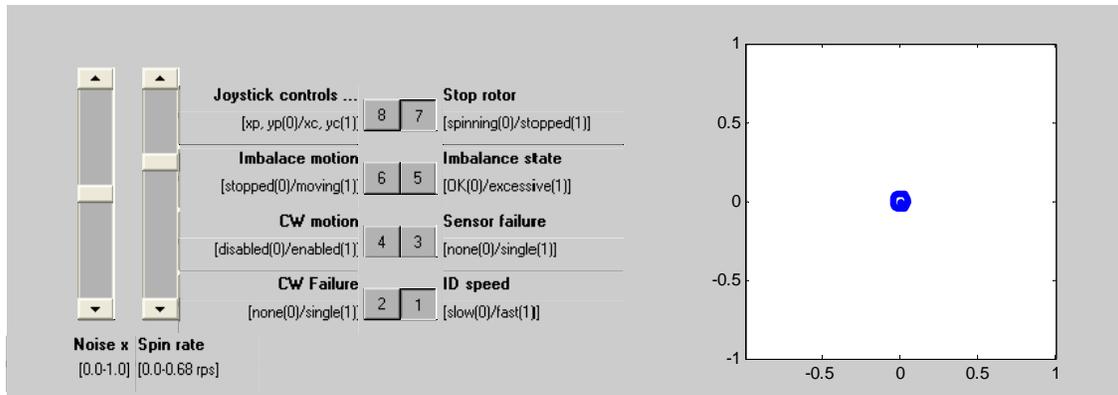
American Institute of Aeronautics and Astronautics

Figure 11.   Software simulation GUI – simulation control panel

This control panel is used to control the simulation, starting/stopping the rotor spin, counterweights, and simulated random imbalance motion. Sensor and counterweight failures are also controlled. The imbalance may be "driven" manually (rather than the pseudo-random walk) by using the mouse to drag the blue circle joystick emulator on the right.
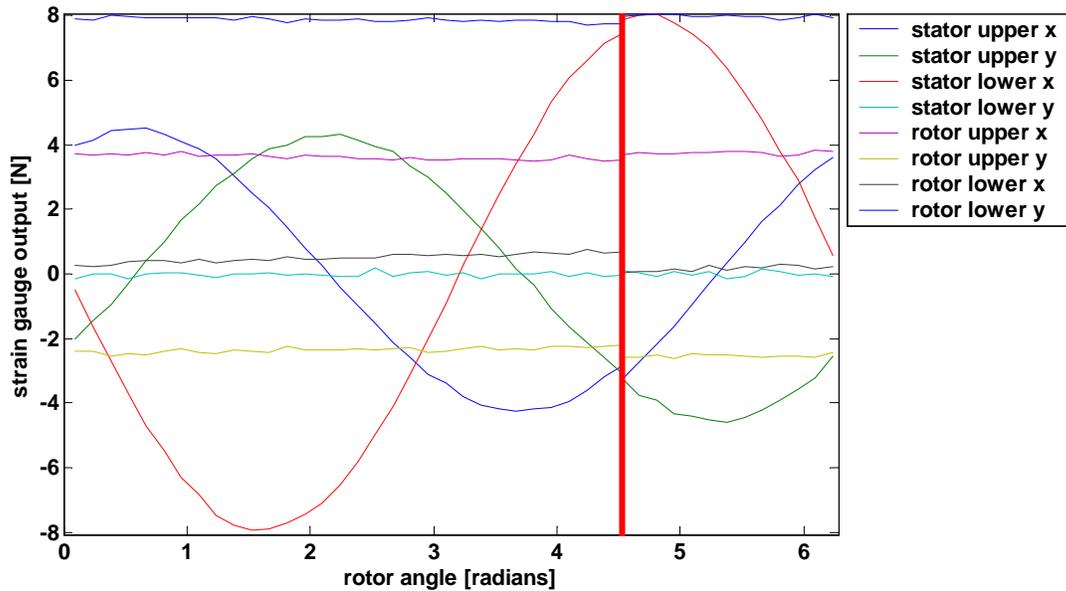

Figure 12.   Software simulation GUI – strain gauge outputs

The 8 strain gauge signals are shown, plotted against the rotor angle. The vertical red line shows the beginning/end of the most recent revolution. If the imbalance were not moving, there were no vibration or sensor noise, and no failures were present, the 4 rotating gauges would read constant values and the 4 fixed gauges would have sinusoidal values (2 pairs 90 degrees in phase apart). However, "stator lower y" has failed, resulting in a reading of zero + noise.

VI. Conclusion

Algorithms that provide automatic balancing and autonomous fault tolerance for a space-based centrifuge have been derived and successfully implemented in software simulation. Although developed for an Earth-based simulator, the underlying principles may be extended for application to the ISS Centrifuge design which has a 5-axis vibration isolation mechanism suspension and displacement sensors.

The architecture, which is similar to an indirect adaptive control architecture, allows careful and independent tuning of identification and control bandwidths, which is a key challenge for this application.

The sensor FDIR method presented uses maximum likelihood theory and is integrated closely with the imbalance identification algorithms, facilitating reconfiguration in the case of a sensor fault. This method does not rely on the rotor dynamical model, and is independent of rotor disturbances, relying only on the self-consistency of the redundant sensor suite.

The overall architecture and FDIR algorithms should be applicable to other aerospace systems that require automatic, fault tolerant, on-line identification and control of system properties.

Appendix

A. Abbreviations

BDS:	Bearing displacement sensor
CM:	Center of mass
CW:	Counterweight
EOM:	Equation(s) of motion
FDI:	Fault detection and isolation
FDIR:	Fault detection isolation and reconfiguration (or recovery)
FIR:	Finite impulse response (filter)
GUI:	Graphical user interface
Hz:	Hertz, cycles per second
ID:	Identification (system identification)
IID:	Independent and identically distributed
ISS:	International Space Station
JAXA:	Japan Aerospace Exploration Agency
KF:	Kalman Filter
LVLH:	Local vertical local horizontal
LS:	Least squares
m:	Meters
MCI:	Mass-couple imbalance
N:	Newtons
NASA:	National Aeronautics and Space Administration
NFN:	Net Force Noise
PMI:	Point-mass imbalance
RBNB:	Ring-buffered network bus
RLS:	Recursive least squares
SFN:	Strain gauge Force Noise
SG:	Strain gauge
SNR:	Signal to noise ratio
SSN:	Strain gauge Sensor Noise
SSBRP:	Space Station Biological Research Project
SSRL:	Smart Systems Research Lab
UZGRV:	Uncorrelated zero-mean Gaussian random variable
VCM:	Voice coil motor
VIM:	Vibration Isolation Mechanism

B. Recursive least squares algorithm

The identification algorithms make extensive use of recursive least squares estimation. This appendix presents the standard algorithm used, as published in Ref. 8, along with certain modifications made to increase run-time performance and simplicity at a slight expense of loss of accuracy. The following is paraphrased from Ref. 8, starting at p.375.

The system of equations is

$$Y = \Phi\theta + \varepsilon \tag{123}$$

Note that differs from the $Ax = b + \varepsilon$ nomenclature used in the report. So $Y \to b, \Phi \to A, \theta \to x, \varepsilon \to -\varepsilon$

Let m_θ and m_Φ indicate the number of rows in the $\hat{\theta}$ vector, and Φ matrix, respectively. Then

- $m_\Phi > m_\theta$ so that the problem is over determined and a least squares fit will be chosen.

- Y is a $m_\Phi \times 1$ vector, generally containing the measurements.

- Φ is a $m_\Phi \times m_\theta$ vector, generally known parameters and some measurements (but ideally, should contain only known quantities).

- θ is a $m_\theta \times 1$ vector of the parameters to be identified.

- ε is a $m_\Phi \times 1$ vector, containing noise.

The batch least squares solution to this is
$$\hat{\theta}_{LS} = (\Phi^T \Phi)^{-1} \Phi^T Y . \tag{124}$$

The weighted batch least squares solution to this is
$$\hat{\theta}_{WLS} = (\Phi^T W \Phi)^{-1} \Phi^T W Y . \tag{125}$$

In the recursive least squares implementation, the rows in the matrix Eq. (123) are stepped through one at a time as follows (algorithm follows Ref. 8 almost exactly):
1. Select a and γ.
2. Comment: $a = \gamma = 1$ is ordinary least squares; $a = 1 - \gamma$ and $0 < \gamma < 1$ is exponentially weighted least squares.
3. Select values for $\hat{\theta}_0$ and P_0, the initial estimates for the parameters and their estimate error covariance.
4. Calculate $\phi_1^T \triangleq \Phi(1,:)$, the first row in the Φ matrix. ϕ_1^T is a $1 \times m_\theta$ row vector taken from Φ, ϕ_1 is a column vector, keeping with the standard notation used in Ref. 8.
5. $k = 0$
6. $L_{k+1} = \dfrac{P_k}{\gamma} \phi_{k+1} \left(\dfrac{1}{a} + \phi_{k+1}^T \dfrac{P_k}{\gamma} \phi_{k+1} \right)^{-1}$ (note that the inverse is of a scalar; L is a $m_\theta \times 1$ column vector, effectively acting as the gain for $\hat{\theta}$ updates)
7. Calculate $y_{k+1} \triangleq Y(k+1)$ (y is a scalar)
8. $\hat{\theta}_{k+1} = \hat{\theta}_k + L_{k+1} \left(y_{k+1} - \phi_{k+1}^T \hat{\theta}_k \right)$ ($\hat{\theta}$ is a $m_\theta \times 1$ column vector)
9. $P_{k+1} = \dfrac{1}{\gamma} \left(I - L_{k+1} \phi_{k+1}^T \right) P_k$ (P is a $m_\theta \times m_\theta$ matrix of the covariances in the identification parameter errors. However, it is based only upon P_0 and the rows of Φ (but not Y), so as far as using it as a measure of convergence, it reflects the information input, but not the closeness of fit. Analysis of the residuals calculated in step 8 may be more meaningful for that.)
10. Calculate $\phi_{k+2}^T \triangleq \Phi(k+2,:)$, the next row in the Φ matrix.
11. $k = k + 1$
12. Go to step 6.

Modification 1

If the regression equation changes so that each row of the Φ matrix is used by multiple θ vectors to produce multiple y scalars, the dimensions of the variables change as follows:

- Y is a $m_\Phi \times n_\theta$ matrix.

- θ is a $m_\theta \times n_\theta$ matrix of the parameters to be identified.

- ε is a $m_\Phi \times n_\theta$ matrix, containing noise.

This effectively combines the multiple RLS IDs (for each column in θ) into one regression. Each ID is really completely independent, but since they share the same Φ matrix, they can be calculated together (as long as they can also share P_0, as discussed below). If this is the situation, the previous equations can be used exactly, except that y becomes a $1 \times n_\theta$ row vector (and the above changes are made as well). The limitation that comes with this convenience is that the columns of θ share the same P matrix. One P_0 matrix of size $m_\theta \times m_\theta$ initializes P, and then, since each column of θ uses the same row of Φ, the P matrix is appropriately updated. So the only real limitation is that each column of θ should be able to share a common P_0. If this does not hold, then different P matrices should be propagated for each column of θ, effectively meaning each column is identified independently.

Modification 2 (applicable in this case)

In a second permutation on the original, covered in Eq. (8.70 – 8.72) of Ref. 8, the regression equation could change so that the θ vector is shared by multiple rows of the Φ matrix, producing a vector of measurements, y, at each sample. Assuming m_y measurements are taken at each sample, the dimensions of the variables change as follows:

- m_Φ is now greater by a factor of m_y. It is equal to m_y times the number of times at which samples are taken. This increases the number of rows in Y, Φ, and ε.

There are three approaches that can be taken in this case:
Approach 1: (exact) As presented in Eq. (8.70 – 8.72) of Ref. 8, the original algorithm is modified as follows:

- a becomes a diagonal matrix, so a^{-1} is needed instead of $\dfrac{1}{a}$ in step 6, meaning that a matrix inversion is now required at each update. The inversion is of a square matrix with dimension equal to the number of measurements at each sample update.

- ϕ_k^T becomes a $m_y \times m_\theta$ matrix of row vectors in steps 4, 6, 8, 9, and 10.

- L becomes a $m_\theta \times m_y$ gain matrix

Approach 2: Run the original implementation one row at a time, ignoring the fact that m_y samples come in at the same time (the approach taken here). γ should be adjusted to account for the fact that the RLS ID will be updated m_y times for each sample update. The major benefit over Approach 1 is that the matrix inverse is avoided.

Approach 3: Same as Approach 2 except that the order of updating flips on each measurement update. So on the odd sample updates, the RLS is updated with rows $1, 2, \cdots, m_y$ taken one at a time, in that order. On the even updates, the RLS is updated with rows $m_y, m_y - 1, \cdots, 2, 1$.

These different approaches give identical results if $a = \gamma = 1$ (no exponential weighting) since in that case, the sequence of measurements does not matter. For example, at the extreme, it does not matter if the measurements come in one at a time or all at once, as long as Y and Φ are the same. However, for the case where exponential weighting is applied, these approaches give different results.

If γ is adjusted as mentioned above, the first two approaches will differ slightly, since in Approach 2, the $m_y{}^{th}$ measurement will be weighted slightly more than the first measurement in each sample update (since it appears to be more recent, even though it is not). If γ has been chosen to provide a relatively slow exponential decay (so the approximate sample window is much greater than m_y), this difference that Approach 2 gives is probably acceptable given the benefit of avoiding the matrix inverse. Approach 2 would be consistently biased towards weighting the last elements in the measurement vector higher than the first elements.

Approach 3 is aimed at reducing that potential problem in a way that still avoids the matrix inverse. In that case, the ordering of the RLS updates will flip on each sample update. The following plot shows the effective weightings applied to samples for the 3 different approaches. The weightings have been normalized so that the average weighting across all measurements in each Approach is 1.0. In this example, 10 measurements come in at each sample update. For the exact method, each of the 10 measurements taken at one time in each batch is weighted equally, as shown by the horizontal blue segments. In Approach 2, the weightings decrease exponentially going back in time/sequence (some of the Approach 2 points are covered with Approach 3 points).

References

[1]Otsuki, F., Uematsu, H., Nakamura, Y., Ohtomi, K., Chida, Y., Kawamoto, O., "Approach to Realization of Micro-gravity Performance of Centrifuge Rotor System," Copyright 1998 Society of Automotive Engineers, Inc.

[2]Otsuki, F., Uematsu, H., Nakamura, Y., Chida, Y., Nishimura, O., Ohtomi, K., and Tanaka, M., "Vibration Isolation Control of Centrifuge Rotor," *5th International Conference on Motion and Vibration Control*, 2000, pp. 415-420.

[3]Ohtomi, K., Otsuki, F., Uematsu, H., Nakamura, Y., Chida, Y., and Kawamoto, O., "Approach to Realization of Micro-gravity Performance of Centrifuge Rotor," *30th International Conference on Environmental Systems Proceedings*, 2000.

[4]Ohtomi, K., Otsuki, F., Uematsu, H., Nakamura, Y., Chida, Y., Nishimura, O., and Okamura, R., "Active Mass Auto-balancing System for Centrifuge Rotor Providing an Artificial Gravity in Space," *31st International Conference on Environmental Systems Proceedings*, 2001.

[5]Ohtomi, K., "Centrifuge rotor integrated analysis," *Modeling, Simulation and Calibration of Space-based Systems, SPIE Vol. 5420*, April 2004.

[6]Lawson, C. and Hanson, R., *Solving Least Squares Problems*, Series in Automatic Computation, Prentice Hall, Englewood Cliffs, NJ, 1974.

[7]Gelb, A, *et al.*, *Applied Optimal Estimation*, MIT Press, Cambridge, MA, 1974.

[8]Franklin, G., Powell, J.D., and Workman, M.L., *Digital Control of Dynamic Systems,* Second Edition, Addison Wesley Publishing, Menlo Park, California, 1990.

[9]Devore, J.L., *Probability and Statistics for Engineering and the Sciences,* Second Edition, Brooks/Cole Publishing Company, Belmont, California, 1987.

[10]MATLAB is a registered trademark of The MathWorks, Inc., 24 Prime Park Way, Natick, MA 01760, 508-647-7000.