SSRL Centrifuge Autobalancing and FDIR

(work in progress)

Edward Wilson, Ph.D., P.E.[1]

Robert W. Mah, Ph.D.[2]

Smart Systems Research Laboratory

NASA Ames Research Center

25 July 2003

Abstract

This report summarizes centrifuge-related work performed at the Smart Systems Research Laboratory at NASA Ames Research Center's Computational Sciences Division from 1995 through 2003. The goal is to develop an automated system that will sense an imbalance (both static and dynamic[3]) in a centrifuge and issue control commands to drive counterweights to eliminate the effects of the imbalance. This autobalancing development began when the ISS centrifuge design was not yet finalized, and was designed to work with the SSRL Centrifuge laboratory prototype, constructed in 1993-1995 and shown in Figure 1. Significant differences between that prototype and the current International Space Station (ISS) Centrifuge design are that:

- The spin axis for the SSRL Centrifuge prototype can translate freely in x and y, but not wobble, whereas the ISS centrifuge spin axis has 3 translational and two rotational degrees of freedom, supported by a vibration isolation mechanism.

- The imbalance sensors 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 resulting from imbalance.

High fidelity autobalancing and FDIR systems (for both counterweights and strain gauges) are developed and tested in MATLAB simulation, for the SSRL Centrifuge configuration. Hardware implementation of the autobalancing technology was begun in 1996, but was terminated due to lack of funding. The project lay dormant until 2001-2002 when the FDIR capability was added.

Now that the ISS centrifuge configuration is (close to) finalized, the logical extension of this research is to extend its application to the new configuration to evaluate the potential autobalancing performance improvements and FDIR capabilities. Following are the steps towards completing this task, as of 25 July 2003.

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. Update to use the present counterweight locations.

4. Update to use the present bearing displacement sensors (BDSs) - this would seem to force me to undo the step where all sensors are first reduced to a net force and torque on the rotor

---

[1] Ed.Wilson@intellization.com, 454 Barkentine Lane, Redwood Shores, CA 94065-1126

[2] Robert.W.Mah@nasa.gov, SSRL group lead

[3] Here, in accordance with standard rotor-balancing terminology, a "static" imbalance is one that creates a net force or torque on the centrifuge rotor in the presence of gravity when the rotor is not spinning (i.e. the center of mass is not on the axis of rotation). This is the reason for the name "static", although there will be no imbalance force created in zero g with the rotor stopped. A "dynamic" imbalance creates a net force or torque on the centrifuge rotor when the rotor has a non-zero angular velocity or angular acceleration (i.e., the cross-axis terms in the inertia matrix are non-zero).

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

6. Integrate with ADAMS for testing – thought is to have a simpler, possibly linearized 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

Table of Contents

Table of Figures

1 Introduction

1.1 SSRL Centrifuge laboratory prototype

The SSRL Centrifuge laboratory prototype is shown in Figure 1. It was developed in 1993-1995 by Robert Mah and Mike Guerrero at the SSRL, with Alessandro Galvagni assisting with the computer systems interfaces. It floats 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. There is slight axial motion permitted (see below), but it is a much stiffer suspension than the VIM of the ISS Centrifuge.
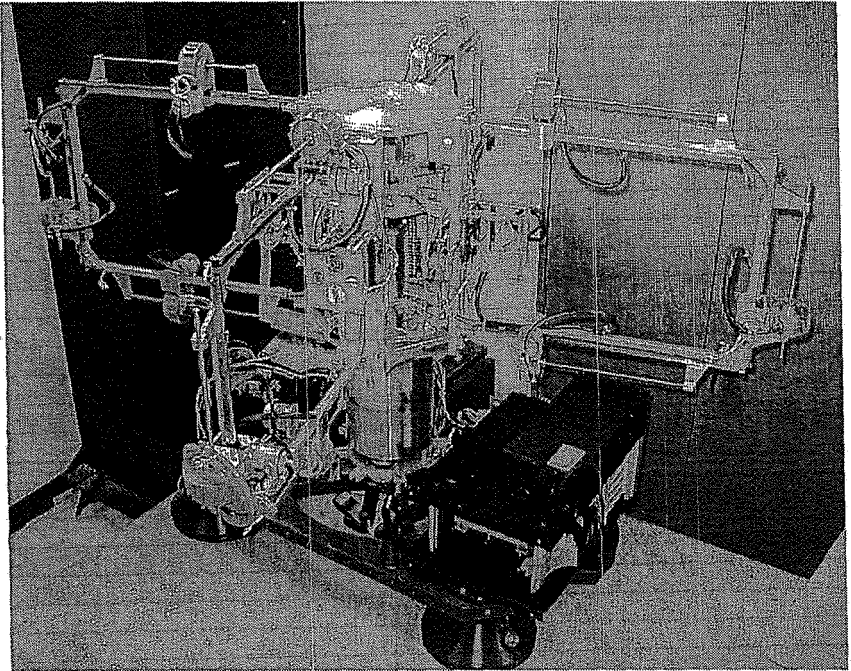
Figure 1: The SSRL Centrifuge Autobalancing Hardware Simulator

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.

There are locations for 24 strain gauges (SGs) to measure imbalance forces, 8 do not rotate, and are for measuring the force between the stator and the air bearing platform. No SGs are installed at these locations. 8 SGs rotate with the rotor, near the upper CW plane. Of these 8 locations, 4 SGs are installed, equally spaced through one rotation. 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 visco-elastic, allowing for slight displacement (on the order of 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.

1.2 Brief summary of the autobalancing method presented in this report:

1. Centrifuge rotor is modeled as a rigid body spinning about a fixed axis.

2. Imbalance is represented in a compact, intuitive way as the x- and y-locations of a point mass in the 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 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 prior approach was developed by the author, concluding in October 1995, that was very similar. The major improvement made in this updated 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.

2   Control System Architecture
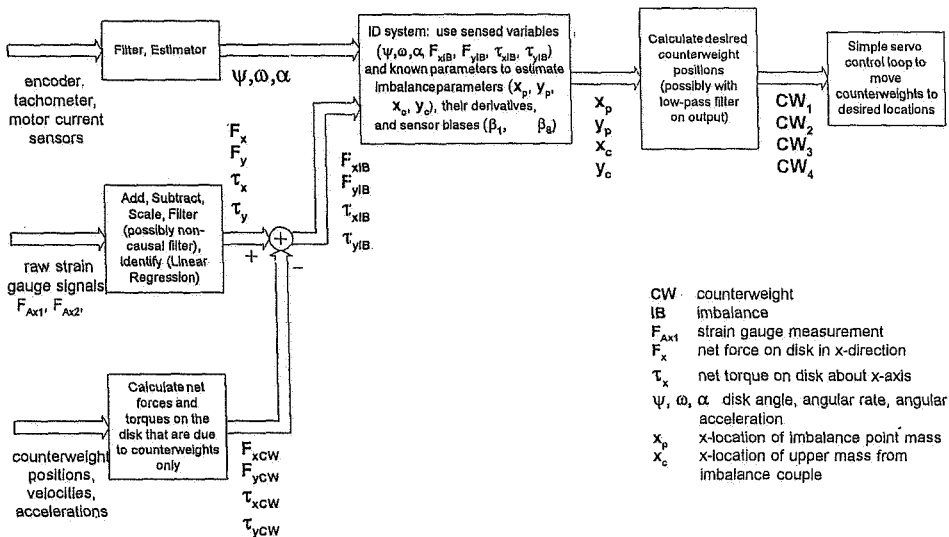
Figure 2: Control system block diagram

3   Imbalance Identification

The basic approach taken here to identify the imbalance 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.

3.1   Description of a structure to model a general state of imbalance in the rotor (defined by a set of "model-imbalance parameters").

The first step here is to select a general form for the imbalance that is capable of representing any possible imbalance. The 3x3 inertia matrix describing the spun portion of the centrifuge (the "rotor") could be used, but a more intuitive representation is presented here. This arbitrarily chosen imbalance model structure has been proven to be sufficiently general to model any possible imbalance.[4]

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.

---

[4]See Appendix C

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 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)[5] The net forces and torques $[F_x, F_y, \tau_x, \tau_y]$ on the rotor will be calculated as functions of the rotor motion (ψ, ω, α), fixed system parameters (g, 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.
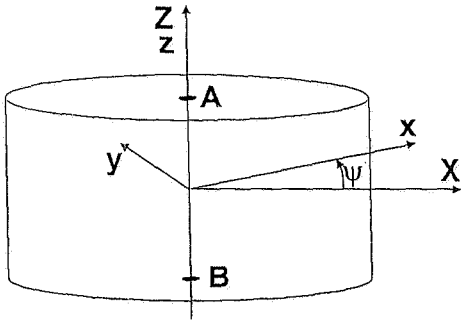
x, y, z are fixed in the disk frame
X, Y, Z are fixed on the de-spun portion of the centrifuge
ψ(t) is the angle of the disk the only rotationa degree of freedom
A, B represent the upper and lower portions of the disk
z = 0 at the midpoint between the counterweight planes

Figure 3: Coordinate system used to describe imbalances and their effects

The coordinate system is shown in Figure 3, with the z-axis locations shown in detail in Figure 4 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[6] 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 complete list of variables is contained in Appendix A

---

[5]The rotor angular acceleration force, Coriolis force, and force due to imbalance acceleration are expected to be minimal, but will be included for completeness.
[6]Although the centrifuge base may actually rotate slightly, this will be a small effect, and it is neglected in this analysis.
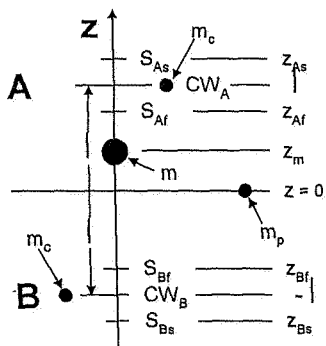
A (B) refers to the upper (lower) part of the disk

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 4: 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 4, are defined arbitrarily to lie in the same planes as the counterweights. The "point-mass" imbalance is chosen arbitrarily to lie in the plane 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_{Bf}, and z_{Bs}, as shown in Figure 4.

To summarize, the physical locations of the counterweight planes define the locations of the z-axis and the model imbalances (PMI and MCI).
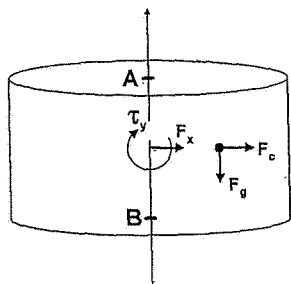
F_a is the force on the point masses due to angular acceleration (not shown in the figures)

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.
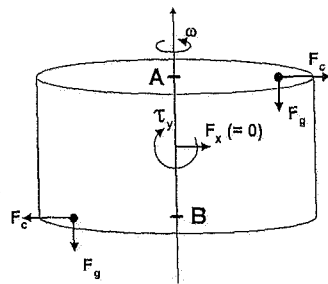
Figure 5: Model imbalances

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

Figure 5 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. This is proven in Appendix C and summarized as follows. The proof 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, $m_p x_p = \Sigma_{i=1}^n m_i x_i$, where m_i and x_i represent the mass and x-coordinate of each of the many point masses). In Section 3.2, the effective net forces and torques on the rotor due to the model imbalances will be calculated.

3.2   Derivation of equations relating model-imbalance parameters to the net forces and torques on the rotor.

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

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:

- F_g, the force due to gravity. This acts straight down at all times[7] 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.

$$F_x = 0 \qquad\qquad F_y = 0 \qquad\qquad \tau_x = -m_p g y_p \qquad\qquad \tau_y = m_p g x_p$$

- 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 $m_p r_p \omega^2$

$$F_x = m_p \omega^2 x_p \qquad\qquad F_y = m_p \omega^2 y_p \qquad\qquad \tau_x = 0 \qquad\qquad \tau_y = 0$$

- 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 $m_p r_p \alpha$.

$$F_x = m_p \alpha y_p \qquad\qquad F_y = -m_p \alpha x_p \qquad\qquad \tau_x = 0 \qquad\qquad \tau_y = 0$$

- 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.

$$F_x = 2m_p \omega v_{yp} \qquad\qquad F_y = -2m_p \omega v_{xp} \qquad\qquad \tau_x = 0 \qquad\qquad \tau_y = 0$$

- 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}).

$$F_x = -m_p a_{xp} \qquad\qquad F_y = -m_p a_{yp} \qquad\qquad \tau_x = -m_p a_{zp} y_p \qquad\qquad \tau_y = m_p a_{zp} x_p$$

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

---

[7]Effects due to misalignment between the gravity vector and the axis of rotation are described in Section 3.2.1

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.

- 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.

$$F_x = 0 \qquad\qquad F_y = 0 \qquad\qquad \tau_x = 0 \qquad\qquad \tau_y = 0$$

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

$$F_x = 0 \qquad\qquad F_y = 0 \qquad\qquad \tau_x = -2m_c\omega^2 y_c \qquad\qquad \tau_y = 2m_c\omega^2 x_c$$

- 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.

$$F_x = 0 \qquad\qquad F_y = 0 \qquad\qquad \tau_x = 2m_c\alpha x_c \qquad\qquad \tau_y = 2m_c\alpha y_c$$

- 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.

$$F_x = 0 \qquad\qquad F_y = 0 \qquad\qquad \tau_x = 4m_c\omega v_{xc} \qquad\qquad \tau_y = 4m_c\omega v_{yc}$$

- 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}).

$$F_x = 0 \qquad\qquad F_y = 0 \qquad\qquad \tau_x = 2m_c a_{yc} - 2m_c a_{zc} y_c \qquad\qquad \tau_y = -2m_c a_{xc} + 2m_c a_{zc} x_c$$

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

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

Putting the equations in a linear form so that the linear regression can be applied later requires dropping the "cross-product" terms. These are some of the terms due to acceleration of the imbalance within the rotor frame likely to be small.

$F_x =$	$m_p\omega^2 x_p$	$+ m_p\alpha y_p$	$+ 2m_p\omega v_{yp}$	$m_p a_{xp}$	
$F_y =$	$m_p\omega^2 y_p$	$m_p\alpha x_p$	$2m_p\omega v_{xp}$	$m_p a_{yp}$	
$\tau_x =$	$-m_p g y_p$	$2m_c\omega^2 y_c$	$+ 2m_c\alpha x_c$	$+ 4m_c\omega v_{xc}$	$+ 2m_c a_{yc}$
$\tau_y =$	$m_p g x_p$	$+ 2m_c\omega^2 x_c$	$+ 2m_c\alpha y_c$	$+ 4m_c\omega v_{yc}$	$2m_c a_{xc}$

3.2.1 Additional forces

The misalignment between the rotor axis of rotation and the gravity vector will cause strain gauge measurements that are not due to the imbalances. 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. This misalignment causes a constant force on the fixed strain gauges and an oscillating forces on the spinning strain gauges.

The misalignment causes forces due to the model imbalances 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.
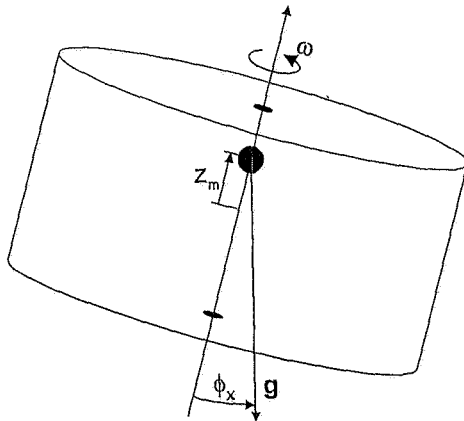
Figure 6: Misalignment between the axis of rotation and the gravity vector results in additional forces that are not due to imbalances

$F_{g\phi}$, 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 gravity (c.o.g.), 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.

$$F_X = mg\phi_x \qquad F_Y = mg\phi_y \qquad \tau_X = -mg\phi_y z_m \qquad \tau_Y = mg\phi_x z_m$$

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 $-\psi$.

$$F_x = F_X\cos\psi + F_Y\sin\psi \qquad F_y = -F_X\sin\psi + F_Y\cos\psi \qquad \tau_x = \tau_X\cos\psi + \tau_Y\sin\psi \qquad \tau_y = -\tau_X\sin\psi + \tau_Y\cos\psi$$

Substituting.

$$F_x = mg\phi_x \cos\psi + mg\phi_y \sin\psi \qquad\qquad F_y = -mg\phi_x \sin\psi + mg\phi_y \cos\psi$$

$$\tau_x = -mg\phi_y z_m \cos\psi + mg\phi_x z_m \sin\psi \qquad\qquad \tau_y = mg\phi_y z_m \sin\psi + mg\phi_x z_m \cos\psi$$

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 $z_{m\phi x}$ ($= z_m \phi_x$) and $z_{m\phi y}$ ($= z_m \phi_y$) are introduced to maintain linearity.

$$F_x = mg\phi_x \cos\psi + mg\phi_y \sin\psi \qquad\qquad F_y = -mg\phi_x \sin\psi + mg\phi_y \cos\psi$$

$$\tau_x = -mg\, z_{m\phi y} \cos\psi + mg\, z_{m\phi x} \sin\psi \qquad\qquad \tau_y = mg\, z_{m\phi y} \sin\psi + mg\, z_{m\phi x} \cos\psi$$

Adding these forces and torques to the model-imbalance forces and torques,

$$F_x = \quad m_p\omega^2 x_p \quad + m_p\alpha y_p \quad + 2m_p\omega v_{yp} \quad m_p a_{xp} \quad + mg\cos\psi\,\phi_x + mg\sin\psi\,\phi_y$$

$$F_y = \quad m_p\omega^2 y_p \qquad m_p\alpha x_p \qquad 2m_p\omega v_{xp} \qquad m_p a_{yp} \qquad mg\sin\psi\,\phi_x + mg\cos\psi\,\phi_y$$

$$\tau_x = \quad -m_p g y_p \qquad 2m_c\omega^2\!/y_c + 2m_c\alpha/x_c + 4m_c\omega/v_{xc} + 2m_c/a_{yc} + mg\sin\psi\,z_{m\phi x} \quad mg\cos\psi\,z_{m\phi y}$$

$$\tau_y = \quad m_p g x_p \qquad + 2m_c\omega^2\!/x_c + 2m_c\alpha/y_c + 4m_c\omega/v_{yc} \qquad 2m_c/a_{xc} + mg\cos\psi\,z_{m\phi x} + mg\sin\psi\,z_{m\phi y}$$

Putting these equations in matrix form,

$$F_{netB} = \Phi\,\theta$$

where θ is a 16x1 column vector of the model-imbalance parameters. m, m_p, and m_c may be chosen arbitrarily, since they simply scale the values of the other parameters. They are chosen to equal 1 (kilogram) so they drop out of the equations completely. Φ is a 4x16 matrix. ($s\psi \equiv \sin\psi$, $c\psi \equiv \cos\psi$).

$\theta =$

| x_p | y_p | x_c | y_c | v_{xp} | v_{yp} | v_{xc} | v_{yc} | a_{xp} | a_{yp} | a_{xc} | a_{yc} | ϕ_x | ϕ_y | $z_{m\phi x}$ | $z_{m\phi y}$ |

$\Phi =$

ω^2	α	0	0	0	2ω	0	0	-1	0	0	0	$g\,c\psi$	$g\,s\psi$	0	0
$-\alpha$	ω^2	0	0	-2ω	0	0	0	0	-1	0	0	$-g\,s\psi$	$g\,c\psi$	0	0
0	$-g$	$2\alpha f$	$-2\omega^2 f$	0	0	$4\omega f$	0	0	0	0	$2f$	0	0	$g\,s\psi$	$-g\,c\psi$
g	0	$2\omega^2 f$	$2\alpha f$	0	0	0	$4\omega f$	0	0	$-2f$	0	0	0	$g\,c\psi$	$-g\,s\psi$

3.2.2 Summary

The above linear matrix equation describes the net forces and torques on the rotor, $F_{netB} = [F_x, F_y, \tau_x, \tau_y]^T$ that are due to the model-imbalance parameters, $\theta = [x_p, y_p, x_c, y_c, v_{xp}, v_{yp}, v_{xc}, v_{yc}, a_{xp}, a_{yp}, a_{xc}, a_{yc}, \phi_x, \phi_y, z_{m\phi x}, z_{m\phi y}]^T$ which represent the model imbalances as well as the axis misalignment $[\phi_x, \phi_y, z_{m\phi x}, z_{m\phi y}]$. Note that the MCI produces torques only no forces, and the PMI produces no torques except for the gravity effects.

Section 3.4 presents a linear regression method to estimate the model-imbalance parameters based upon force and torque measurements. However, sensors to measure $[F_x, F_y, \tau_x, \tau_y]$ directly are not available, so before this method can be applied, these signals must first be estimated based upon the strain gauge measurements.

3.3 Derivation of equations relating counterweight parameters to the net forces and torques on the rotor.

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 2b.

Problem definition:

Counterweight parameters, m_{pCW}, m_{cCW}, and $\theta_{CW} = [x_{pCW}, y_{pCW}, x_{cCW}, y_{cCW}, v_{xpCW}, v_{ypCW}, v_{xcCW}, v_{ycCW}, a_{xpCW}, a_{ypCW}, a_{xcCW}, a_{ycCW}]^T$ define the counterweight properties that result in net forces and torques on the rotor These parameters are defined with the same structure as used for the model imbalances.

Take measurements of counterweight positions, velocities, and accelerations, and calculate the above parameters. Then use these parameters to calculate the net forces and torques applied to the rotor by the counterweights.

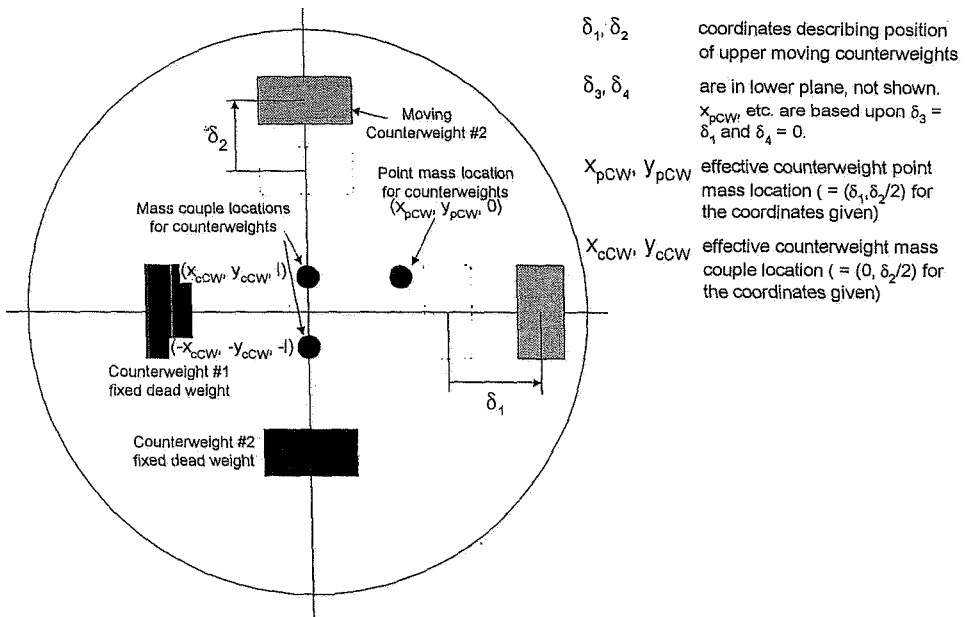This is to be performed at each sample.

On the figure, labels read:

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

$-\delta_2$

δ_1

Figure 7: Counterweight coordinates

The counterweights are arranged in the layout shown in Figure 7. The positions of each of the four counterweights is described by $[\delta_1, \delta_2, \delta_3, \delta_4]$. 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 counterweight is exactly counterbalanced by the dead weight.

The velocities and accelerations are described by the first and second derivatives of the position signals, $[\dot{\delta}_1 \ \dot{\delta}_2 \ \dot{\delta}_3 \ \dot{\delta}_4 \ \ddot{\delta}_1 \ \ddot{\delta}_2 \ \ddot{\delta}_3 \ \ddot{\delta}_4]$. Sensors will not be used to measure these signals directly (i.e., no 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 for completeness. For this analysis, the counterweight coordinate vector, $[\delta_1, \delta_2, \delta_3, \delta_4, \dot{\delta}_1 \ \dot{\delta}_2 \ \dot{\delta}_3 \ \dot{\delta}_4 \ \ddot{\delta}_1 \ \ddot{\delta}_2 \ \ddot{\delta}_3 \ \ddot{\delta}_4]$, is assumed to be available at all times.

The first step is to calculate the counterweight parameters $\theta_{CW} = [x_{pCW}, y_{pCW}, x_{cCW}, y_{cCW}, v_{xpCW}, v_{ypCW}, v_{xcCW}, v_{ycCW}, a_{xpCW}, a_{ypCW}, a_{xcCW}, a_{ycCW}]^T$ from the counterweight coordinates, $[\delta_1, \delta_2, \delta_3, \delta_4, \dot{\delta}_1 \ \dot{\delta}_2 \ \dot{\delta}_3 \ \dot{\delta}_4 \ \ddot{\delta}_1 \ \ddot{\delta}_2 \ \ddot{\delta}_3 \ \ddot{\delta}_4]$. The basic approach is similar to the proof shown in Appendix C — add the individual effects of each of the counterweights and select the counterweight parameters to yield the same effects. The equations are simplified if the counterweights all have mass, equal to m_{CW} (this is the moving portion of the counterweights).

$m_{pCW} = 4\, m_{CW}$

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

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

$v_{xpCW} = d/dt(x_{pCW}) = (\dot{\delta}_1 + \dot{\delta}_3) / 4$

$v_{ypCW} = d/dt(y_{pCW}) = (\dot{\delta}_2 + \dot{\delta}_4) / 4$

$$a_{xpCW} = d/dt(v_{xpCW}) = (\delta_1'' + \delta_3'')/4$$

$$a_{ypCW} = d/dt(v_{ypCW}) = (\delta_2'' + \delta_4'')/4$$

$$m_{cCW} = 2\,m_{CW}$$

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

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

$$v_{xcCW} = d/dt(x_{cCW}) = (\delta_1' - \delta_3')/4$$

$$v_{ycCW} = d/dt(y_{cCW}) = (\delta_2' - \delta_4')/4$$

$$a_{xcCW} = d/dt(v_{xcCW}) = (\delta_1'' - \delta_3'')/4$$

$$a_{ycCW} = d/dt(v_{ycCW}) = (\delta_2'' - \delta_4'')/4$$

Substitute these counterweight parameters into the model-imbalance equations:

$$F_{xCW} = m_{pCW}\omega^2 x_{pCW} + m_{pCW}\alpha y_{pCW} + 2m_{pCW}\omega v_{ypCW} \quad m_{pCW}a_{xpCW}$$

$$F_{yCW} = -m_{pCW}\alpha x_{pCW} + m_{pCW}\omega^2 y_{pCW} \quad 2m_{pCW}\omega v_{xpCW} \quad m_{pCW}a_{ypCW}$$

$$\tau_{xCW} = m_{pCW}g y_{pCW} + 2m_{cCW}\alpha \ell x_{cCW} \quad 2m_{cCW}\omega^2 \ell y_{cCW} + 4m_{cCW}\omega \ell v_{xcCW} + 2m_{cCW}\ell a_{ycCW}$$

$$\tau_{yCW} = m_{pCW}g x_{pCW} + 2m_{cCW}\omega^2 \ell x_{cCW} + 2m_{cCW}\alpha \ell y_{cCW} + 4m_{cCW}\omega \ell v_{ycCW} \quad 2m_{cCW}\ell a_{xcCW}$$

Dropping terms including counterweight velocity and acceleration:

$$F_{xCW} = m_{pCW}\omega^2 x_{pCW} + m_{pCW}\alpha y_{pCW}$$

$$F_{yCW} = -m_{pCW}\alpha x_{pCW} + m_{pCW}\omega^2 y_{pCW}$$

$$\tau_{xCW} = m_{pCW}g y_{pCW} + 2m_{cCW}\alpha \ell x_{cCW} \quad 2m_{cCW}\omega^2 \ell y_{cCW}$$

$$\tau_{yCW} = m_{pCW}g x_{pCW} + 2m_{cCW}\omega^2 \ell x_{cCW} + 2m_{cCW}\alpha \ell y_{cCW}$$

Putting these equations in matrix form,

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

where θ_{CW} is a 4x1 column vector of the model-imbalance parameters. Φ_{CW} is a 4x4 matrix

$\theta_{CW} =$	$[x_{pCW}$	y_{pCW}	x_{cCW}	$y_{cCW}]^T$
$\Phi_{CW} =$	$m_{pCW}\omega^2$	$m_{pCW}\alpha$	0	0
	$m_{pCW}\alpha$	$m_{pCW}\omega^2$	0	0
	0	$m_{pCW}g$	$2\,m_{cCW}\alpha\ell$	$-2\,m_{cCW}\omega^2\ell$
	$m_{pCW}g$	0	$2\,m_{cCW}\omega^2\ell$	$2\,m_{cCW}\alpha\ell$

3.4 Identification of the net forces and torques based upon the strain gauge signals.

The goal in this section is to develop a procedure to produce net force and torque measurements, $[F_x, F_y, \tau_x, \tau_y]$, at each sample period based upon strain gauge measurements. 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 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:

16 strain gauges are used: 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" indicate the axis of measurement. "f" or "s" (not shown here) indicate "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.
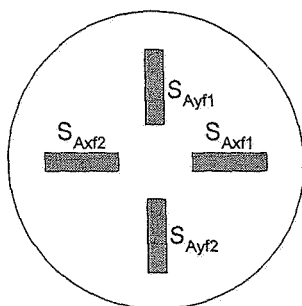

Figure 8: Strain gauge layout

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 +10N and S_{Axf2} will be in compression, reading -10N. The combined value, S_{Axf}, will be $(10 - (-10))/2 = 10N$. This combined value will also be filtered to reduce sensor noise and extraneous vibrations. For now, this operation is represented as "filter()"

$$S_{Axf} = filter((S_{Axf1} - S_{Axf2}) / 2)$$

$$S_{Ayf} = filter((S_{Ayf1} - S_{Ayf2}) / 2)$$

$$S_{Bxf} = filter((S_{Bxf1} - S_{Bxf2}) / 2)$$

$$S_{Byf} = filter((S_{Byf1} - S_{Byf2}) / 2)$$

$$S_{Axs} = filter((S_{Axs1} - S_{Axs2}) / 2)$$

$$S_{Ays} = filter((S_{Ays1} - S_{Ays2}) / 2)$$

$$S_{Bxs} = filter((S_{Bxs1} - S_{Bxs2}) / 2)$$

$$S_{Bys} = filter((S_{Bys1} - S_{Bys2}) / 2)$$

Assume that this force measurement is composed of three parts:

$$S = S_{true} + \beta + \varepsilon$$

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:

$$S = S_{true} + B + E_{SSN}$$

where,

S is an 8x1 column vector $\quad [S_{Axf}, S_{Ayf}, S_{Bxf}, S_{Byf}, S_{Axs}, S_{Ays}, S_{Bxs}, S_{Bys}]^T$

S_{true} is an 8x1 column vector: $\quad [S_{Axf\,true}, S_{Ayf\,true}, S_{Bxf\,true}, S_{Byf\,true}, S_{Axs\,true}, S_{Ays\,true}, S_{Bxs\,true}, S_{Bys\,true}]^T$

B is an 8x1 column vector $\quad [\beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6, \beta_7, \beta_8]^T$

E_{SSN} is an 8x1 column vector $\quad [\varepsilon_{SS1}, \varepsilon_{SS2}, \varepsilon_{SS3}, \varepsilon_{SS4}, \varepsilon_{SS5}, \varepsilon_{SS6}, \varepsilon_{SS7}, \varepsilon_{SS8}]^T$ ("$_{SSN}$" indicates Strain gauge Sensor Noise)

Assume that S_{true}, the actual force at the strain gauge, can be directly calculated from the net forces and torques on the rotor, $[F_x, F_y, \tau_x, \tau_y]$, 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). Putting this in equation form:

$$S_{true} = T\, G\, F_{net} + E_{SFN}$$

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 $z_{Bs} / (z_{Bs} - z_{As})$. "G" is for "geometry."

F_{net} is the 4x1 vector of net forces and torques on the rotor, $[F_x, F_y, \tau_x, \tau_y]$.

E_{SFN} is an 8x1 column vector $\quad [\varepsilon_{SF1}, \varepsilon_{SF2}, \varepsilon_{SF3}, \varepsilon_{SF4}, \varepsilon_{SF5}, \varepsilon_{SF6}, \varepsilon_{SF7}, \varepsilon_{SF8}]^T$ ("$_{SFN}$" indicates Strain gauge Force Noise)

To first derive G, assume the rotor angle, ψ, is zero so T is the identity matrix. In this case,

$$S_{true} = G\, F_{net}$$

$S_{Axf\,true} =$	G_{11}	G_{12}	G_{13}	G_{14}	F_x
$S_{Ayf\,true}$	G_{21}	G_{22}	G_{23}	G_{24}	F_y
$S_{Bxf\,true}$	G_{31}	G_{32}	G_{33}	G_{34}	τ_x
$S_{Byf\,true}$	G_{41}	G_{42}	G_{43}	G_{44}	τ_y
$S_{Axs\,true}$	G_{51}	G_{52}	G_{53}	G_{54}	
$S_{Ays\,true}$	G_{61}	G_{62}	G_{63}	G_{64}	
$S_{Bxs\,true}$	G_{71}	G_{72}	G_{73}	G_{74}	
$S_{Bys\,true}$	G_{81}	G_{82}	G_{83}	G_{84}	

Each of the elements in G will now be identified by force and torque balance equations.

$$F_x = S_{Axf\,true} + S_{Bxf\,true}$$

$$\tau_y = S_{Axf\,true}\, z_{Af} + S_{Bxf\,true}\, z_{Bf}$$

solving these two equations with two unknowns,

$$S_{Axf\,true} = [-z_{Bf} / (z_{Af} \cdot z_{Bf})] F_x + [1 / (z_{Af} \cdot z_{Bf})] \tau_y$$

$$S_{Bxf\,true} = [z_{Af} / (z_{Af} \cdot z_{Bf})] F_x + [-1 / (z_{Af} \cdot z_{Bf})] \tau_y$$

Repeating for y,

$$F_y = S_{Ayf\,true} + S_{Byf\,true}$$

$$\tau_x = -S_{Ayf\,true}\, z_{Af} + -S_{Byf\,true}\, z_{Bf}$$

solving these two equations with two unknowns,

$$S_{Ayf\,true} = [-z_{Bf} / (z_{Af} \cdot z_{Bf})] F_y + [-1 / (z_{Af} \cdot z_{Bf})] \tau_x$$

$$S_{Byf\,true} = [z_{Af} / (z_{Af} \cdot z_{Bf})] F_y + [1 / (z_{Af} \cdot z_{Bf})] \tau_x$$

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

$$S_{Axs\,true} = [-z_{Bs} / (z_{As} \cdot z_{Bs})] F_x + [1 / (z_{As} \cdot z_{Bs})] \tau_y$$

$$S_{Bxs\,true} = [z_{As} / (z_{As} \cdot z_{Bs})] F_x + [-1 / (z_{As} \cdot z_{Bs})] \tau_y$$

$$S_{Ays\,true} = [-z_{Bs} / (z_{As} \cdot z_{Bs})] F_y + [-1 / (z_{As} \cdot z_{Bs})] \tau_x$$

$$S_{Bys\,true} = [z_{As} / (z_{As} \cdot z_{Bs})] F_y + [1 / (z_{As} \cdot z_{Bs})] \tau_x$$

So the G matrix is:

$$
G = \begin{bmatrix}
-z_{Bf}/(z_{Af}\ z_{Bf}) & 0 & 0 & 1/(z_{Af}\cdot z_{Bf}) \\
0 & -z_{Bf}/(z_{Af}\cdot z_{Bf}) & -1/(z_{Af}\cdot z_{Bf}) & 0 \\
z_{Af}/(z_{Af}\ z_{Bf}) & 0 & 0 & -1/(z_{Af}\cdot z_{Bf}) \\
0 & z_{Af}/(z_{Af}\ z_{Bf}) & 1/(z_{Af}\ z_{Bf}) & 0 \\
-z_{Bs}/(z_{As}\ z_{Bs}) & 0 & 0 & 1/(z_{As}\ z_{Bs}) \\
0 & -z_{Bs}/(z_{As}\ z_{Bs}) & -1/(z_{As}\ z_{Bs}) & 0 \\
z_{As}/(z_{As}\ z_{Bs}) & 0 & 0 & -1/(z_{As}\ z_{Bs}) \\
0 & z_{As}/(z_{As}\ z_{Bs}) & 1/(z_{As}\ z_{Bs}) & 0
\end{bmatrix}
$$

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

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

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,

$$
\begin{bmatrix}
S_{Axf\,true} \\
S_{Ayf\,true} \\
S_{Bxf\,true} \\
S_{Byf\,true}
\end{bmatrix}
=
\begin{bmatrix}
\cos\psi & -\sin\psi & 0 & 0 \\
\sin\psi & \cos\psi & 0 & 0 \\
0 & 0 & \cos\psi & -\sin\psi \\
0 & 0 & \sin\psi & \cos\psi
\end{bmatrix}
\begin{bmatrix}
S_{Axf\,true}|_{\psi=0} \\
S_{Ayf\,true}|_{\psi=0} \\
S_{Bxf\,true}|_{\psi=0} \\
S_{Byf\,true}|_{\psi=0}
\end{bmatrix}
$$

So the T matrix is,

$$
T = \begin{bmatrix}
\cos\psi & -\sin\psi & 0 & 0 & 0 & 0 & 0 & 0 \\
\sin\psi & \cos\psi & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

0	0	$\cos\psi$	$-\sin\psi$	0	0	0	0
0	0	$\sin\psi$	$\cos\psi$	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

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

$T^{-1} =$	$\cos\psi$	$\sin\psi$	0	0	0	0	0	0
	$-\sin\psi$	$\cos\psi$	0	0	0	0	0	0
	0	0	$\cos\psi$	$\sin\psi$	0	0	0	0
	0	0	$-\sin\psi$	$\cos\psi$	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

It is simple to check that $T^{-1}T = I$.

Now that G and T have been derived, equations *** and *** are combined to yield,

$$S = T\,G\,F_{net} + B + E_{SFN} + E_{SSN}$$

Rearranging,

$$S - B = T\,G\,F_{net} + (E_{SFN} + E_{SSN})$$

$$T^{-1}(S - B) = G\,F_{net} + T^{-1}(E_{SFN} + E_{SSN})$$

This is now in the standard form for a least squares problem: $y = Ax + e$, where $T^{-1}(S-B)$ is y, G is A, F_{net} is x, and $T^{-1}(E_{SFN} + E_{SSN})$ is e (still white noise). The least-squares solution for this equation is $x = (A^T A)^{-1} A^T y$. The former representation, with (S-B) as y and TG as A, could have been used; however, by pre-multiplying by T^{-1} the A matrix remains constant, so it (and $(A^T A)^{-1} A^T$) does not need to be re-calculated at each update. The least-squares minimization will be derived here for this problem.

Problem statement:

Assume a system governed by the above equation. T^{-1} and G are known.[8] S is measured, but is corrupted by the bias and noise terms as shown. Find $\hat{F_{net}}$ that can reproduce $(T^{-1}(S-B))\hat{}$ by the equation $(T^{-1}(S-B))\hat{} = G\,\hat{F_{net}}$ where the error $((T^{-1}(S-B))\hat{} - (T^{-1}(S-B)))^2$ is minimized. The idea is that by finding $\hat{F_{net}}$ that minimizes this cost function will be close to the actual F_{net}, and this problem formulation is mathematically easy to solve.

$$\text{minimize over } \hat{F_{net}} \quad J = ((T^{-1}(S-B))\hat{} - (T^{-1}(S-B)))^2$$

---

[8] 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

$$J = (G\ F_{net}\hat{}\quad (T^{-1}(S-B)))^2$$

the cost J is minimized when $\partial J/\partial Fnet\hat{} = 0$

$$\partial J/\partial Fnet\hat{} = 2\ \partial/\partial Fnet\hat{}(G\ F_{net}\hat{}\quad (T^{-1}(S-B)))^T\ (G\ F_{net}\hat{}\quad (T^{-1}(S-B)))$$

$$\partial J/\partial Fnet\hat{} = 2\ G^T\ (G\ F_{net}\hat{}\quad (T^{-1}(S-B)))$$

setting the derivative, $\partial J/\partial Fnet\hat{} = 0$,

$$0 = 2\ G^T\ (G\ F_{net}\hat{}\quad (T^{-1}(S-B)))$$

$$0 = G^T\ G\ F_{net}\hat{}\quad G^T(T^{-1}(S-B))$$

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

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

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

Summarizing,

$$\mathbf{F_{net}}\hat{} = \mathbf{\Gamma\ T}^{-1}\mathbf{(S-B)}$$

The term, $((G^T\ G)^{-1}\ G^T)$ is a fixed 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 $\Gamma \equiv ((G^T\ G)^{-1}\ G^T)$. Some of the elements of Γ are always zero, as indicated below. Also, $\Gamma_{22} = \Gamma_{11}, \Gamma_{24} = \Gamma_{13}, \Gamma_{26} = \Gamma_{15}, \Gamma_{28} = \Gamma_{17}, \Gamma_{32} = -\Gamma_{41}, \Gamma_{34} = -\Gamma_{43}, \Gamma_{36} = -\Gamma_{45}, \Gamma_{38} = -\Gamma_{47}.$

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 calculated at each sample.

S is the 8x1 vector of measurements resulting from the filtering and combination of strain gauge pairs.

B is the 8x1 bias vector that will be calculated in the following identification.

$s\psi \equiv \sin\psi$, $c\psi \equiv \cos\psi$ in the equation below.

$F_x\hat{}$ =	Γ_{11}	0	Γ_{13}	0	Γ_{15}	0	Γ_{17}	0	$c\psi$	$s\psi$	0	0	0	0	0	0	S_{Axf} β_1
$F_y\hat{}$	0	Γ_{22}	0	Γ_{24}	0	Γ_{26}	0	Γ_{28}	$-s\psi$	$c\psi$	0	0	0	0	0	0	S_{Ayf} β_2
$\tau_x\hat{}$	0	Γ_{32}	0	Γ_{34}	0	Γ_{36}	0	Γ_{38}	0	0	$c\psi$	$s\psi$	0	0	0	0	S_{Bxf} β_3
$\tau_y\hat{}$	Γ_{41}	0	Γ_{43}	0	Γ_{45}	0	Γ_{47}	0	0	0	$-s\psi$	$c\psi$	0	0	0	0	S_{Byf} β_4
									0	0	0	0	1	0	0	0	S_{Axs} β_5
									0	0	0	0	0	1	0	0	S_{Ays} β_6
									0	0	0	0	0	0	1	0	S_{Bxs} β_7
									0	0	0	0	0	0	0	1	S_{Bys} β_8

This concludes the identification of the net forces and torques on the rotor. $F_{net}\hat{} = [F_x\hat{}\ F_y\hat{}\ \tau_x\hat{}\ \tau_y\hat{}]$, based on the strain gauge measurements.

3.5 Calculation of forces due to imbalances alone by subtracting out the effects of the counterweights.

The net forces and torques on the rotor are due to the sum of the imbalance forces and the counterweight forces:

$$F_{net} = F_{netIB} + F_{netCW}$$

$F_{net}\hat{}$ has been estimated based upon the strain gauge measurements, and F_{netCW} has been calculated based upon the counterweight coordinates. Subtracting the two leaves F_{netIB}, the forces and torques due to the imbalances.

$$F_{netlB} = F_{net}^{\wedge} \quad F_{netCW}$$

expanding,

$$F_{netlB} = \Gamma\, T^{-1}(S-B)\quad F_{netCW}$$

where,

$$
\begin{bmatrix} F_x^{\wedge} \\ F_y^{\wedge} \\ \tau_x^{\wedge} \\ \tau_y^{\wedge} \end{bmatrix}
=
\begin{bmatrix}
\Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\
0 & \Gamma_{22} & 0 & \Gamma_{24} & 0 & \Gamma_{26} & 0 & \Gamma_{28} \\
0 & \Gamma_{32} & 0 & \Gamma_{34} & 0 & \Gamma_{36} & 0 & \Gamma_{38} \\
\Gamma_{41} & 0 & \Gamma_{43} & 0 & \Gamma_{45} & 0 & \Gamma_{47} & 0
\end{bmatrix}
\begin{bmatrix}
c\psi & s\psi & 0 & 0 & 0 & 0 & 0 & 0 \\
-s\psi & c\psi & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & c\psi & s\psi & 0 & 0 & 0 & 0 \\
0 & 0 & -s\psi & c\psi & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix} S_{Axf} \\ S_{Ayf} \\ S_{Bxf} \\ S_{Byf} \\ S_{Axs} \\ S_{Ays} \\ S_{Bxs} \\ S_{Bys} \end{bmatrix}
\begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \\ \beta_6 \\ \beta_7 \\ \beta_8 \end{bmatrix}
$$

$$F_{xCW} = m_{pCW}\omega^2 x_{pCW} \quad + m_{pCW}\alpha y_{pCW} \quad + 2m_{pCW}\omega v_{ypCW} \quad m_{pCW}a_{xpCW}$$

$$F_{yCW} = m_{pCW}\omega^2 y_{pCW} \quad m_{pCW}\alpha x_{pCW} \quad 2m_{pCW}\omega v_{xpCW} \quad m_{pCW}a_{ypCW}$$

$$\tau_{xCW} = -m_{pCW}g y_{pCW} \quad 2m_{cCW}\omega^2 / y_{cCW} \quad + 2m_{cCW}\alpha / x_{cCW} \quad + 4m_{cCW}\omega / v_{xcCW} \quad + 2m_{cCW}/a_{ycCW}$$

$$\tau_{yCW} = m_{pCW}g x_{pCW} \quad + 2m_{cCW}\omega^2 / x_{cCW} \quad + 2m_{cCW}\alpha / y_{cCW} \quad + 4m_{cCW}\omega / v_{ycCW} \quad 2m_{cCW}/a_{xcCW}$$

3.6 Identification of the model-imbalance parameters based upon these net forces and torques.

From the previous step,

$$F_{netlB} = \Gamma\, T^{-1}(S-B)\quad F_{netCW}$$

From Section 3.3,

$$F_{netlB} = \Phi\, \theta$$

Combining these equations, a least squares minimization problem can be solved to obtain the model-imbalance parameters, $\theta = [x_p, y_p, x_c, y_c, v_{xp}, v_{yp}, v_{xc}, v_{yc}, a_{xp}, a_{yp}, a_{xc}, a_{yc}, \phi_x, \phi_y, z_{m\phi x}, z_{m\phi y}]^T$

F_{netlB} will actually have a noise signal added to the forces and torques resulting from the imbalance parameters, so,

$$F_{netlB} = \Phi\, \theta + E_{NFN}$$

E_{NFN} is an 8x1 column vector $[\varepsilon_{NF1}, \varepsilon_{NF2}, \varepsilon_{NF3}, \varepsilon_{NF4}, \varepsilon_{NF5}, \varepsilon_{NF6}, \varepsilon_{NF7}, \varepsilon_{NF8}]^T$ ("NFN" indicates Net-Force Noise)

where E_{NFN} is the "Net-Force noise" signal. Earlier noise vectors E_{SFN} and E_{SSN} were for "Strain gauge Force Noise" (forces at the strain gauges that did not result in net forces) and "Strain gauge Sensor Noise" (sensor errors not resulting from actual forces).

Combining equations *** and ***

$$\Gamma\, T^{-1}(S-B)\quad F_{netCW} = \Phi\, \theta + E_{NFN}$$

$$\Gamma\, T^{-1} S - \Gamma\, T^{-1} B \quad F_{netCW} = \Phi\, \theta + E_{NFN}$$

$$\Gamma\, T^{-1} S \quad F_{netCW} = \Phi\, \theta + \Gamma\, T^{-1} B + E_{NFN}$$

$\Gamma T^{-1} S \quad F_{netCW} = [\Phi, \Gamma T^{-1}] [\theta, B] + E_{NFN}$

This is in the same form as the standard linear least squares minimization problem, where $\theta^\wedge$ and $B^\wedge$ will be estimated. The left side of the equation (the usual place for the vector of measurements) can be calculated directly, resulting in a 4x1 vector.

The solution to this equation is:

$$[\theta^\wedge, B^\wedge] = ([\Phi, \Gamma T^{-1}]^T [\Phi, \Gamma T^{-1}])^{-1} [\Phi, \Gamma T^{-1}]^T (\Gamma T^{-1} S - F_{netCW})$$

where (summarizing from throughout this document),

$\theta^\wedge$ is the estimated model-imbalance parameter vector.

$$\theta^\wedge = [x_p^\wedge \; y_p^\wedge \; x_c^\wedge \; y_c^\wedge \; v_{xp}^\wedge \; v_{yp}^\wedge \; v_{xc}^\wedge \; v_{yc}^\wedge \; a_{xp}^\wedge \; a_{yp}^\wedge \; a_{xc}^\wedge \; a_{yc}^\wedge \; \phi_x^\wedge \; \phi_y^\wedge \; z_{m\phi x}^\wedge \; z_{m\phi y}^\wedge]^T$$

$B^\wedge$ is the estimated strain gauge bias vector,

$$B^\wedge = [\beta_1^\wedge \; \beta_2^\wedge \; \beta_3^\wedge \; \beta_4^\wedge \; \beta_5^\wedge \; \beta_6^\wedge \; \beta_7^\wedge \; \beta_8^\wedge]^T$$

Φ is the matrix defining the forces and torques resulting from the imbalance parameters,

$\Phi =$

ω^2	α	0	0	0	2ω	0	0	-1	0	0	0	g cψ	g sψ	0	0
-α	ω^2	0	0	-2ω	0	0	0	0	1	0	0	-g sψ	g cψ	0	0
0	-g	$2\alpha f$	$-2\omega^2 f$	0	0	$4\omega f$	0	0	0	0	$2f$	0	0	g sψ	-g cψ
g	0	$2\omega^2 f$	$2\alpha f$	0	0	0	$4\omega f$	0	0	$-2f$	0	0	0	g cψ	g sψ

Γ is a fixed 4x8 matrix that is a function of the fixed system geometry only (strain gauge locations, etc.). It is calculated once only. Some of the elements of Γ are always zero, as indicated below. Also, $\Gamma_{22} = \Gamma_{11}, \Gamma_{24} = \Gamma_{13}, \Gamma_{26} = \Gamma_{15}, \Gamma_{28} = \Gamma_{17}, \Gamma_{32} = -\Gamma_{41}, \Gamma_{34} = -\Gamma_{43}, \Gamma_{36} = -\Gamma_{45}, \Gamma_{38} = -\Gamma_{47}$.

$$
\Gamma = \quad
\begin{matrix}
\Gamma_{11} & 0 & \Gamma_{13} & 0 & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\
0 & \Gamma_{22} & 0 & \Gamma_{24} & 0 & \Gamma_{26} & 0_7 & \Gamma_{28} \\
0 & \Gamma_{32} & 0 & \Gamma_{34} & 0 & \Gamma_{36} & 0 & \Gamma_{38} \\
\Gamma_{41} & 0 & \Gamma_{43} & 0 & \Gamma_{45} & 0 & \Gamma_{47} & 0
\end{matrix}
$$

$\Gamma \equiv ((G^T G)^{-1} G^T)$

where,

$$
G = \quad
\begin{matrix}
-z_{Bf} / (z_{Af} \cdot z_{Bf}) & 0 & 0 & 1 / (z_{Af} \cdot z_{Bf}) \\
0 & -z_{Bf} / (z_{Af} \cdot z_{Bf}) & 1 / (z_{Af} \cdot z_{Bf}) & 0 \\
z_{Af} / (z_{Af} \cdot z_{Bf}) & 0 & 0 & 1 / (z_{Af} \cdot z_{Bf}) \\
0 & z_{Af} / (z_{Af} \cdot z_{Bf}) & 1 / (z_{Af} \cdot z_{Bf}) & 0 \\
-z_{Bs} / (z_{As} \cdot z_{Bs}) & 0 & 0 & 1 / (z_{As} \cdot z_{Bs}) \\
0 & -z_{Bs} / (z_{As} \cdot z_{Bs}) & 1 / (z_{As} \cdot z_{Bs}) & 0 \\
z_{As} / (z_{As} \cdot z_{Bs}) & 0 & 0 & 1 / (z_{As} \cdot z_{Bs}) \\
0 & z_{As} / (z_{As} \cdot z_{Bs}) & 1 / (z_{As} \cdot z_{Bs}) & 0
\end{matrix}
$$

T^{-1} is an 8x8 matrix with eight terms that change at each sample   all simple functions of ψ.

$$T^{-1} = \begin{bmatrix} \cos\psi & \sin\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ -\sin\psi & \cos\psi & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \cos\psi & \sin\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & -\sin\psi & \cos\psi & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$\Gamma\, T^{-1}$ is a 4x8 matrix with that can be analytically calculated to save run-time computation as follows.

$$\Gamma\, T^{-1} = \begin{bmatrix} \Gamma_{11}\cos\psi & \Gamma_{11}\sin\psi & \Gamma_{13}\cos\psi & \Gamma_{13}\sin\psi & \Gamma_{15} & 0 & \Gamma_{17} & 0 \\ \cdot\Gamma_{11}\sin\psi & \Gamma_{11}\cos\psi & \cdot\Gamma_{13}\sin\psi & \Gamma_{13}\cos\psi & 0 & \Gamma_{15} & 0 & \Gamma_{17} \\ \Gamma_{41}\sin\psi & \cdot\Gamma_{41}\cos\psi & \Gamma_{43}\sin\psi & \cdot\Gamma_{43}\cos\psi & 0 & \cdot\Gamma_{45} & 0 & \cdot\Gamma_{47} \\ \Gamma_{41}\cos\psi & \Gamma_{41}\sin\psi & \Gamma_{43}\cos\psi & \Gamma_{43}\sin\psi & \Gamma_{45} & 0 & \Gamma_{47} & 0 \end{bmatrix}$$

S is the 8x1 vector of measurements resulting from the filtering and combination of strain gauge pairs.

$$S = [S_{Axf}, S_{Ayf}, S_{Bxf}, S_{Byf}, S_{Axs}, S_{Ays}, S_{Bxs}, S_{Bys}]^T$$

S_{Axf} = filter($(S_{Axf1}\; S_{Axf2}) / 2$)

S_{Ayf} = filter($(S_{Ayf1}\; S_{Ayf2}) / 2$)

S_{Bxf} = filter($(S_{Bxf1}\; S_{Bxf2}) / 2$)

S_{Byf} = filter($(S_{Byf1}\; S_{Byf2}) / 2$)

S_{Axs} = filter($(S_{Axs1}\; S_{Axs2}) / 2$)

S_{Ays} = filter($(S_{Ays1}\; S_{Ays2}) / 2$)

S_{Bxs} = filter($(S_{Bxs1}\; S_{Bxs2}) / 2$)

S_{Bys} = filter($(S_{Bys1}\; S_{Bys2}) / 2$)

where S_{Axf1}, etc. are the raw strain gauge signals, and the filter function has yet to be determined.

F_{netCW} is a 4x1 vector of the counterweight forces and torques that is directly calculated at each·sample,

$$F_{netCW} = \begin{bmatrix} m_{pCW}\omega^2 x_{pCW} & + m_{pCW}\alpha y_{pCW} & + 2m_{pCW}\omega v_{ypCW} & m_{pCW}a_{xpCW} \\ m_{pCW}\omega^2 y_{pCW} & m_{pCW}\alpha x_{pCW} & 2m_{pCW}\omega v_{xpCW} & m_{pCW}a_{ypCW} \\ -m_{pCW}g y_{pCW} & 2m_{cCW}\omega^2/y_{cCW} & + 2m_{cCW}\alpha/x_{cCW} & + 4m_{cCW}\omega/v_{xcCW} & + 2m_{cCW}/a_{ycCW} \\ m_{pCW}g x_{pCW} & + 2m_{cCW}\omega^2/x_{cCW} & + 2m_{cCW}\alpha/y_{cCW} & + 4m_{cCW}\omega/v_{ycCW} & 2m_{cCW}/a_{xcCW} \end{bmatrix}$$

This is programmed as a recursive least squares algorithm, so the estimates for θ and B are updated at each sample. B is not used for anything. The last four terms in θ are not used either It is only the first 12 terms in that are used. The extra parameters identified are important since they prevent the useful parameters from being corrupted by the extra forces that they represent.

So, the useful 12 parameters that are identified are,

$$\theta^\wedge = [x_p^\wedge\; y_p^\wedge\; x_c^\wedge\; y_c^\wedge\; v_{xp}^\wedge\; v_{yp}^\wedge\; v_{xc}^\wedge\; v_{yc}^\wedge\; a_{xp}^\wedge\; a_{yp}^\wedge\; a_{xc}^\wedge\; a_{yc}^\wedge]^T$$

These parameters define the position, velocity, and acceleration of the PMI and MCI.

3.7 SG fault detection isolation and reconfiguration (FDIR)

If a SG is lost, F_{net} can still be identified, but not as efficiently as before. The governing equation is repeated here, ignoring B.

$$S = T \, G \, F_{net} + (E_{SFN} + E_{SSN})$$

When a SG is lost (assuming that when it fails it reads zero plus noise — not really the case here, since we're assuming each SG is in a pair reading the exact opposite force), it is as if the corresponding row in the T matrix becomes all zeros. Alternatively, the row can be removed from the T matrix and the S vector, giving them each 7 rows. The choice here is to keep 8 rows. T in this case is not invertible, so the equation cannot be multiplied by T^{-1} as was done before for computational efficiency. So for the least squares solution of F_{net}, $A = T \, G$, $y = S$, and

$$F_{net}{}^{\wedge} = ((G^T \, T^T \, T \, G)^{-1} \, G^T \, T^T) \, S$$

The FDI approach taken here is analogous to the commonly applied bank of Kalman filters. $F_{net}{}^{\wedge}$ is estimated for 9 different conditions: the case of no failures, and then once for each situation where a strain gauge would have failed. The residual from each estimate is then calculated as

$$residual_i = S \quad T_i \, G \, F_{net,i}{}^{\wedge}$$

where i goes from 0 to 8, 0 indicating the case of no failures. T_i is the T matrix corresponding to failure i, and $F_{net,i}{}^{\wedge}$ is the net force estimated assuming failure i is present.

3.8 A MATLAB simulation to validate the identification scheme, including sensitivity to noise in the strain gauge readings

See Appendix B for the MATLAB files. Basic results confirm that noise should not be a significant problem 1 Newton noise 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 120 Hz on a 90 MHz Pentium in MATLAB, 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.
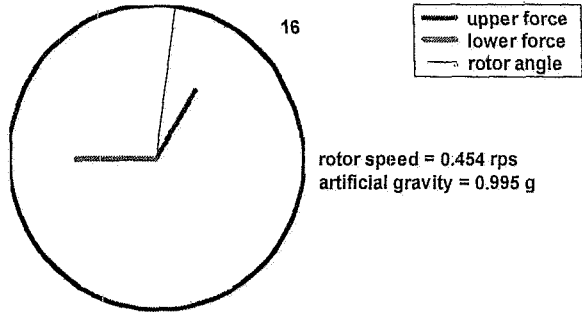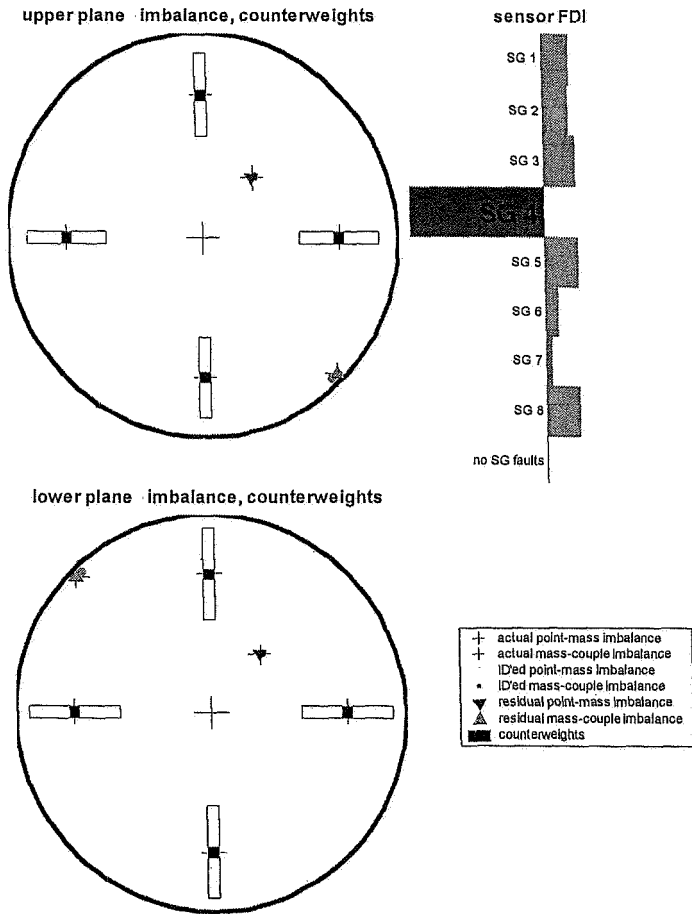
Figure 9: Software simulation GUI - bearing forces

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.

upper plane · imbalance, counterweights

sensor FDI

SG 1
SG 2
SG 3
SG 4
SG 5
SG 6
SG 7
SG 8

no SG faults

lower plane · imbalance, counterweights

+   actual point-mass imbalance
+   actual mass-couple imbalance
·   ID'ed point-mass imbalance
•   ID'ed mass-couple imbalance
▼   residual point-mass imbalance
⚠   residual mass-couple imbalance
▮▮ counterweights

Intellization

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.
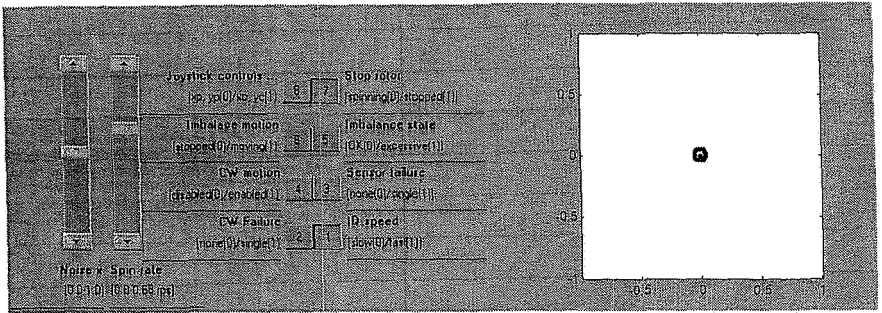
Figure 11: Software simulation GUI – simulation control panel

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.
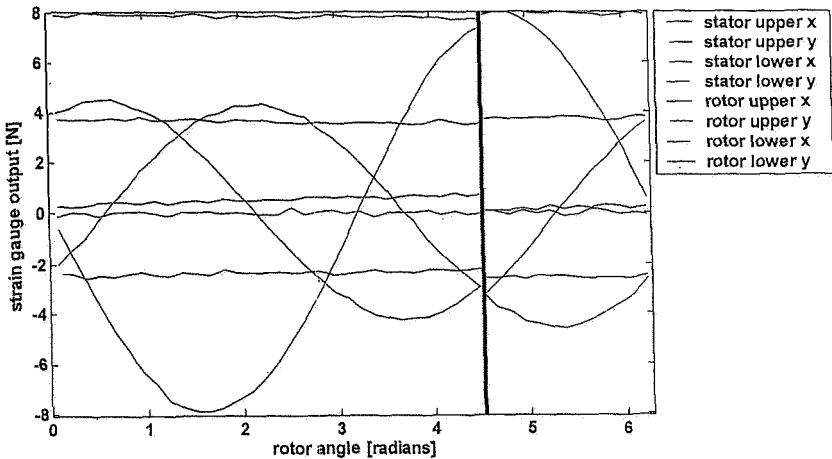

Figure 12: Software simulation GUI – strain gauge outputs

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 was 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). "Stator lower y" has failed, resulting in a reading of zero + noise.

3.9 Filters to produce ψ, ω, and α from the tachometer and encoder signals

How these signals affect the identification:

ψ determines the angular position of the imbalance through the calculation of $\sin\psi$ and $\cos\psi$

ω determines the magnitude of the centrifugal forces calculated. It is especially important since these will be the dominant forces in the absence of gravity, and ω is squared in these calculations.

α determines the magnitude of forces due to angular acceleration. These do not appear to be significant in comparison to the other forces.

If significant angular accelerations were expected, it would make sense to measure the motor current for the rotor motor. Then an estimator could be built that used motor current, angular, and angular-rate sensors along with a model of the rotor drive system to estimate ψ, ω, and α.

However, angular accelerations are expected to be small or steady (during spin-up, for example). The first attempt will be to use simple, independent filters to produce each signal. Analog pre-filters will be used on the encoder and tachometer signals to prevent aliasing. Digital filters will then be used to further condition the signal, and a difference of the filtered tachometer signal will be used for the angular acceleration signal.

3.10 Filters to produce $F_{(A+B)x}$, $F_{(A+B)y}$, $F_{(A-B)x}$, and $F_{(A-B)y}$ from the strain gauge signals

Due to the presence of significant noise, filtering the force signals will be more difficult than filtering the rotor angle, rate, and acceleration signals. The noise in the force signals comes from two major sources: noise due to the sensor itself, and noise due to extraneous vibrations in the centrifuge structure.

It may be useful to look for correlations between the sensors and **subtract** out the vibration signals using adaptive noise cancellation. This should be an excellent research project, and may be difficult. This should be pursued only if more standard methods do not work.

Since these signals are used for identification, and not directly for control, it is possible to use an acausal filter to achieve better performance. Taking this idea to its limit, the data could be collected for the full run (for example, several seconds, resulting in thousands of data points), then it could be filtered forwards and backwards in time (for example, smoothing the data using the filtfilt() function in MATLAB). A modification of this method would be to take data in batches, for example, 2 seconds at a time, smooth the data, and update the ID process once every 2 seconds. To run the ID continuously, avoiding the delay that comes with batch processing, acausal filters could be designed that use 10 samples ahead in time to calculate the measured force. It may be desirable to design these filters adaptively based upon the actual force data.

A possible sequence for this would be:

1) Fix the counterweights and do not move them during this sequence.

2) Directly design reasonably good filters, probably causal ones, to filter the force signals.

3) Run the identification to determine the imbalance parameters and sensor biases.

4) Assume these to be exact and calculate the "actual forces" that would result if they were indeed correct, and there was no noise.

5) Adapt the filters (adding acausal taps if they're not already there) to make the filtered force signals most closely match the "actual forces"

6) Go to step 3) and repeat until the filter weights and identification parameters converge.

The stability of this approach is not known. It's certainly easier to try it out than to try to prove it. During the iterations over steps 3) to 6), the same set of raw data could be used, but this could lead to over-filling -- adaptation to account for the particular data set but failure to generalize. As a final step, allowing the counterweights to move may produce better results. If the effects of the counterweights are found to be significant relative to other noise levels, the effects of their motion could be calculated and subtracted out as part of the force filtering system.

During implementation of the force filtering in general, this is the procedure:

1) Choose a sample rate that is fast enough to detect the signals of interest. The point mass and couple will produce two superimposed sinusoids at the rotation frequency (0.5 to 1.0 Hz). A sample rate of 20 Hz or higher should do well to capture the form of each of these sinusoids. A sample rate of 50 or 100 Hz would be a safe bet.

2) Implement analog pre-filters with break frequencies below the Nyquist frequency, equal to half the sampling frequency So for a 100 Hz sample rate, use 2-pole or better analog pre-filters with break frequencies at 25 Hz.

3) Collect some real data without the counterweights moving.

4) Design causal digital filters (for example, 2-pole Butterworth at 10 Hz), but run them using the MATLAB filtfilt() function (resulting in double the magnitude reduction, but with zero net phase loss the benefit of acausal filtering).

5) Look at how the data is changed by the filter, and tweak the filter parameters until it looks good. If this is not possible, look into ways to tweak other aspects of the force measurement system.

6) Run the identification in batch mode on the filtered data (involving several revolutions of the centrifuge). Tweak it until the ID looks good. Tweak the filters if necessary. At this point, if the imbalance can be identified well, this means that the information is there, and all that remains are (possibly significant) implementation details.

7) Run the identification with 1-2 second updates (perhaps one update per revolution) to see how quickly the ID parameters converge. If it is much slower than 2 seconds, it probably does not make sense to go through the trouble of developing the continuously-updated identification system.

8) If convergence is quick, and development of the continuously updated ID system is warranted, the procedure above for acausal filter design should be followed. In addition to this design issues, communications issues may arise.

Once a structure has been chosen (for example, 2-second batches or acausal filtering with 10-tap leads), it probably makes sense to make the filtering of the rotor angle etc., fit the same structure.

4 Calculation of the desired counterweight positions based upon the identified imbalance

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 3.6.

$$\hat{\theta} = [\hat{x_p}\ \hat{y_p}\ \hat{v_{xp}}\ \hat{v_{yp}}\ \hat{a_{xp}}\ \hat{a_{yp}}\ \hat{x_c}\ \hat{y_c}\ \hat{v_{xc}}\ \hat{v_{yc}}\ \hat{a_{xc}}\ \hat{a_{yc}}]^T$$

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, $[\hat{x_p}\ \hat{y_p}\ \hat{x_c}\ \hat{y_c}]$. If this were accomplished exactly, the velocities and accelerations would match as well.

The counterweight parameters to counteract the imbalance positions are,

$$x_{pCW} = (m_p/m_{pCW})\ \hat{x_p}$$

$$y_{pCW} = (m_p/m_{pCW})\ \hat{y_p}$$

$$x_{cCW} = (m_c/m_{cCW})\ \hat{x_c}$$

$$y_{cCW} = (m_c/m_{cCW})\ \hat{y_c}$$

The equations defining the counterweight parameters are (from Section 3.3),

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

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

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

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

Setting m_p and m_c to 1 kg and combining,

$$(\delta_1 + \delta_3) / 4 = (1/m_{pCW})\ \hat{x_p}$$

$$(\delta_2 + \delta_4) / 4 = (1/m_{pCW})\ \hat{y_p}$$

$$(\delta_1 - \delta_3) / 4 = (1/m_{cCW})\ \hat{x_c}$$

$$(\delta_2 - \delta_4) / 4 = (1/m_{cCW}) \, y_c^{\wedge}$$

Solving for $[\delta_1, \delta_2, \delta_3, \delta_4]$,

$$\delta_1 = -2 \, (\, (x_p^{\wedge}/ m_{pCW}) + (x_c^{\wedge}/ m_{cCW}) \,)$$

$$\delta_2 = -2 \, (\, (y_p^{\wedge}/ m_{pCW}) + (y_c^{\wedge}/ m_{cCW}) \,)$$

$$\delta_3 = -2 \, (\, (x_p^{\wedge}/ m_{pCW}) \;\; (x_c^{\wedge}/ m_{cCW}) \,)$$

$$\delta_4 = -2 \, (\, (y_p^{\wedge}/ m_{pCW}) \;\; (y_c^{\wedge}/ m_{cCW}) \,)$$

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

5   Servo control loop to move the counterweights to their desired locations.

This appears to be handled already by the motor drivers, so unless they are found to be inadequate, they will be used directly.

The desired CW_1, etc. positions will be given directly to the control card. If excessive jitter results from rapidly changing commands, a digital low-pass filter may be added immediately before the command is sent to the motor control card.

6   Development approach / Work plan

Unlike prior work, which involved hardware development, technology development, and software simulation, the proposed work will focus on technology development and software simulation/testing.

The previous work modeled the centrifuge as having a fixed spin axis, representing a hard-mounted design, and matching the SSRL hardware prototype (the SSRL hardware is air-bearing supported, so two axes of translational motion are present, and there is also slight angular motion of the spin axis – but this any deflection is opposed by a stiff spring and angular motion is neglected in the existing model). This simplifies the dynamics by eliminating cross-axis "gyroscopic" coupling, so one of the first steps will be to extend our simulation to involve motion of the spin axis.

The counterweight configuration in the present NASDA design is different from that on our hardware prototype, so our simulation will also be updated to account for the new configuration. The use of displacement sensors, different from the strain gauge sensors used in our hardware, will also be incorporated into the simulation.

Fortunately, the architecture of our autobalancing control system will allow the sensor and counterweight changes to be made fairly easily. The extension of the dynamic modeling to allow spin-axis wobble and translation will be complex, but the control system will be updated to reflect the new equations of motion.

The spin-axis suspension system (the VIM in the NASDA design) will be modeled to represent the VIM as closely as we have accurate information for (e.g., if they release detailed drawings we'll use those, otherwise, the published schematics will allow us to come close). Similarly with the displacement sensor modeling – we will match the NASDA design as closely as possible. The active damping in the VIM is presently designed to act like a viscous damper, so for simplicity, we will model these dampers as passive elements (so the VIM control system does not need to be analyzed).

The NASDA automatic balancing system will be modeled as accurately as possible based on published reports and design review materials.

With the existing NASDA (direct autobalancing control) design and the updated SSRL (indirect autobalancing control) designs integrated into the simulation, we will be able to perform the following specific tests:

1   Experimentally evaluate the robustness of each of these methods to vibrations from the ISS. The results of these tests will address the first proposed research area.
2.   Monitor range of travel for the VIM under various imbalance configurations.
3.   Determine the balancing fidelity possible using both methods with the sensors present in the existing design. If the SSRL approach shows better results, we can quantify that, and this will address the second proposed research area. We can also evaluate what may be possible with different sensors.

Sensor fault detection
Failures in sensors, either displacement- or force-, will impact the autobalancing control, probably resulting in excessive vibration forces. The imbalance should still be observable even with multiple sensor failures, as long as the failed sensors are identified and the autobalancing system is fault tolerant. This work will leverage off other Fault Detection Identification and Recovery (FDIR) research by the SSRL.

Direct vs. indirect autobalancing
The work so far at the SSRL has used an indirect control architecture, whereas the published NASDA approach uses a direct architecture. The indirect approach should be able to provide better ID, and therefore, better cancellation, in the presence of noisy sensors. A simulation-based study would investigate and quantify the differences between the two approaches. This will be more involved than the analysis proposed above, since the control systems will be tested more thoroughly, rather than in a few specific areas.

Active VIM control
The present design uses the voice coil actuators to implement viscous damping. Active control of these actuators may enable better vibration isolation through feed-forward control to null the spring forces (as desired) and by directly dealing with the gyroscopic complexity by torquing about the correct axis (rather than the present, passive approach that provides a torque about the wrong axis, leading to a nutation that will subsequently be damped out).

Hardware implementation
Although initial technology development will be developed and validated using a software simulation, implementation in hardware would further validate the technology. The SSRL Centrifuge could be retrofitted to more closely approximate the ISS Centrifuge design as follows:

- The rotating strain gauges could be replaced with soft springs/dampers to approximate the VIM stiffness/damping (except with the constraint that the translational and rotational stiffnesses are not individually selectable). This would enable the rotor to translate and wobble with respect to the bearing axis – again, not a perfect approximation of the VIM, but this would introduce the gyroscopic physics.
- With the strain gauges removed, non-rotating, non-contact eddy current displacement sensors could be used on a ring attached to the bottom of the rotor. These sensors would emulate the VIM BDSs.
- The counterweights could be reconfigured to more closely approximate the ISS Centrifuge design.

Appendix A. List of variables

Appendix B. MATLAB files

Appendix C. Proof that model structure is general

The imbalance model structure has been proven to be sufficiently general to model any possible imbalance, for a rigid body rotating about a fixed axis.

(hand-written notes)

Basic approach of the proof is to state that any rigid body can be represented by an infinite collection of point masses. Then it is shown that the resulting net forces (centrifugal, accelerative, Coriolis) summed up from these infinite

masses are equal to the resulting forces from the four imbalance parameters when those parameters are chosen correctly.