Ice loading on podded propeller systems: preliminary test results

Rowell, B.; Duffy, D.

Publisher’s version / Version de l'éditeur:
https://doi.org/10.4224/8895185
Laboratory Memorandum; no. LM-2003-31, 2003

National Research Council Canada
Institute for Ocean Technology

Conseil national de recherches Canada
Institut des technologies océaniques

Laboratory Memorandum

LM-2003-31

Ice Loading on Podded Propeller Systems: Preliminary Test Results

B. Rowell and D. Duffy

December 2003

Canada

DOCUMENTATION PAGE

REPORT NUMBER LM-2003-31	**NRC REPORT NUMBER**	**DATE** December 2003
REPORT SECURITY CLASSIFICATION Unclassified		**DISTRIBUTION** Unlimited
TITLE **Ice Loading on Podded Propellers, Preliminary Test Results**		
AUTHOR(S) Barbara Rowell, Dale Duffy		
CORPORATE AUTHOR(S)/PERFORMING AGENCY(S)		
PUBLICATION		
SPONSORING AGENCY(S)		
IMD PROJECT NUMBER 42_959_16		**NRC FILE NUMBER**

KEY WORDS Azimuthing, podded propellers, ice,	**PAGES** 30,App A-F	**FIGS.** 30	**TABLES** 5

SUMMARY

With the increased interest in arctic shipping the number of vessels navigating in the arctic or being constructed with capabilities do so, is rapidly increasing. Thus this report outlines some of the results obtained during the first phase of testing here at the Institute for Ocean Technology (IOT) involving podded propellers in ice, and also details some of the preparation that has been completed for the second phase of testing. The analysis presented includes systematically determining at what instances and relative positions the apparatus experiences its maximum loading. The analysis shows that the maximum and minimum forces on the apparatus occur as one of the propeller blades is entering or exiting the ice respectively. Also, this report outlines the work that has been completed in preparation for the second phase of testing. This includes conducting a temperature sensitivity analysis for the AMTI six component load cells that were used on the forward and aft bearings and the set up for the global dynamometer calibration.

ADDRESS

National Research Council
Institute for Ocean Technology
P. O. Box 12093, Station 'A'
St. John's, Newfoundland, Canada
A1B 3T5

Tel.: (709) 772-5185, Fax: (709) 772-2462

National Research Council Canada
Conseil national de recherches Canada

Institute for Ocean Technology
Institut des technologies océaniques

ICE LOADING ON PODDED PROPELLER SYSTEMS PRELIMINARY TEST RESULTS

LM-2003-31

Barbara Rowell
Dale Duffy

December 2003

ABSTRACT

With the increased interest in arctic shipping the number of vessels navigating in the arctic, or being constructed with capabilities do so, is rapidly increasing. This report outlines some of the results obtained during the first phase of testing here at the Institute for Ocean Technology (IOT) involving podded propellers in ice. It also details some of the preparation that has been completed for the second phase of testing.

The analysis done helped to identify and fine-tune some of the problems associated with the experimentation procedure and with the equipment itself. The analysis presented includes systematically determining at what instances and relative positions the apparatus experiences its maximum loading. The analysis shows that the maximum and minimum forces on the apparatus occur as one of the propeller blades is entering or exiting the ice respectively. It also shows that the forces experienced go through four complete cycles during one revolution of the propeller. This occurred because the propeller that was used for the investigation consisted of four blades.

Also, this report outlines the work that has been completed in preparation for the second phase of testing. One task that this involved was a temperature sensitivity analysis for the AMTI six component load cells that were used on the forward and aft bearings. Tests were done at three different temperatures and the results showed no appreciable change in slope of the applied load versus AMTI voltage, however the value of the Y intercept for the line did vary from temperature to temperature. Also in preparation for the second phase of testing, the global dynamometer had to be calibrated. At the time of writing this report that had not been completed, however this report describes the setup that had taken place in preparation for the calibration.

TABLE OF CONTENTS

INDEX OF TABLES

INDEX OF EQUATIONS

1 INTRODUCTION

As the increase in arctic shipping continues new builds designed to include ice navigation as a part of their expected working year, are adapting different propulsion systems based on their open water characteristics. These systems include podded propulsors, whose ability to direct thrust in any direction greatly improves maneuverability.

An azimuthing podded propulsion system is a unit in which either a motor to drive the propeller is placed inside the pod or is mechanically geared with motor outside the pod, which is itself an azimuthing device. These propulsion systems have been implemented in different types of ships operating in different conditions from ferries and cruise ships to Arctic tankers and icebreakers.[1] They have different operational configurations: pull (tractor) or pusher mode. Since their first appearance, they have continued to grow in size and power. The power ranges reported in the literature vary considerably: from 1530kW [2] to 30000kW [3].

As these technologies have become more and more widespread in application for arctic vessels the regulations for classifying their use has not kept pace. Currently, there is an ongoing effort to update the regulations governing the design of vessels for arctic navigation, including the propulsion systems, by the International Association of Classification Societies. Amongst other things, these requirements give guidance for the design of machinery for operation in polar regions. So far the loads on podded propellers are simply taken as loads on open propellers and they do not include the loads on the pods themselves. This study aims to close the knowledge gap for this type of propulsion systems. A set of experiments has been designed to measure the contact ice loads on a model podded propulsion system. The objective of these tests is to determine a relationship between the ice thickness in which a vessel navigates, and the resulting cyclic loading experienced by the pod, stern bearing and propeller. This relationship would then be used to modify existing shipping regulations pertaining to the machinery design for arctic conditions

2 EXPERIMENTAL MODEL

For this study, the experimental setup shown in Figure 1 has been designed and built at National Research Council's Institute for Ocean Technologies. It consists of a propeller, a pod, a strut and a model stern. The propeller, a similar design to an R-Class vessel's propeller, is 0.30m in diameter and has four blades. The pod unit with a cylindrical cross section is 0.95m long and has 0.17m diameter. The strut has a uniform cross section with a span of 0.45m. The model stern houses the dynamometer that measures global forces and moments acting on the whole podded system. It will also simulate the stern of a hull to a certain extent. Use of a complete model hull would be necessary to model the effects of hull ice interactions more accurately, however, this is not included in the current scope of the work.

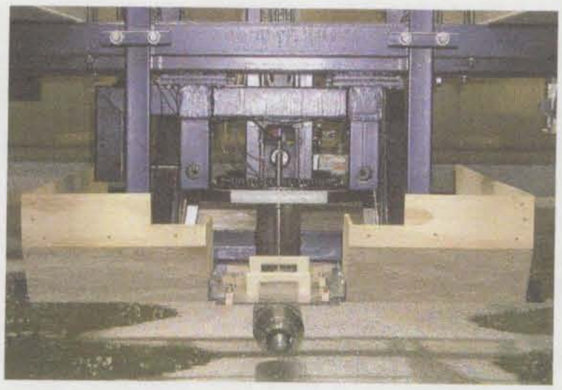


Figure 1. Assembled system (does not including the propellers)

During the tests, four six-component dynamometers are used to measure loads at different locations. These measure loads on a blade, propeller shaft bearing loads, and the global loads on the whole system. Additionally, the propeller shaft is strain gauged near the propeller to measure the torque. The next sections describe these sensors in detail.

2.1 Global Load Balance

Shown in Figure 1 between the carriage mounting frames and the pod is a global load balance. This balance supports the pod and measures the loads that the strut, pod and propeller experiences as a whole in six degrees of freedom. It consists of six high-precision 8907 N load cells oriented as shown in Figure 1. The global dynamometer is fixed in a reference frame attached to the model stern and the carriage. The strut and the pod can be rotated 360 degrees about the vertical axis using the azimuthing gear, while the model stern, and hence the global dynamometer, remains fixed.

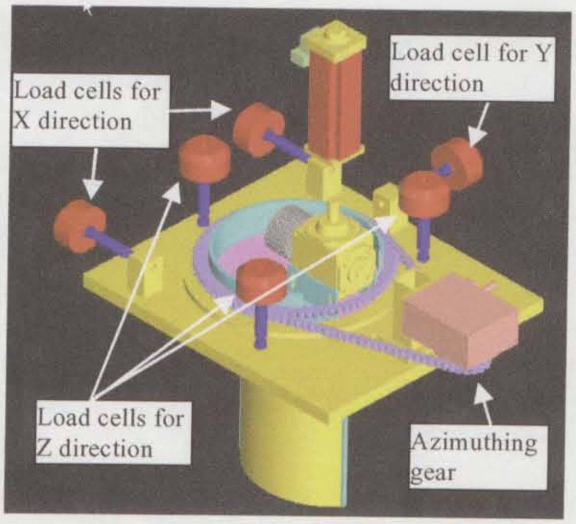


Figure 2. Load cells which for the global load balance

2.2 Blade and Shaft Bearing Dynamometers

The three dynamometers mounted inside the pod, designated as blade, aft and forward dynamometers in Figure 3, are identical and capable of measuring six components: forces and moments in the three orthogonal directions given in Figure 4. The ratings for these dynamometers are as follows:

Forces in x and y directions: 2224 N
Forces in z direction: 4448 N
Moments in x, y and z directions: 56.5 Nm

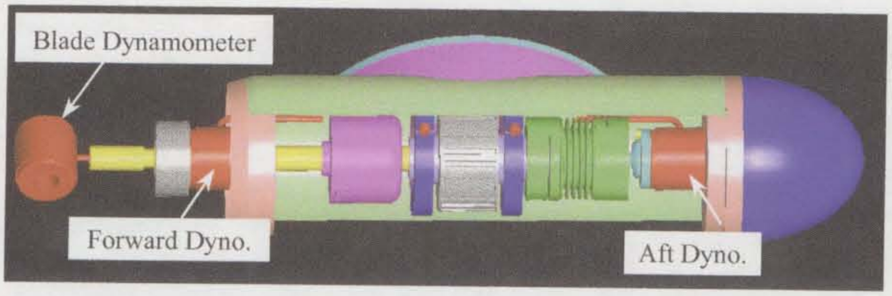


Figure 3. Pod and propeller assembly of the model and orientation of AMTI load cells

The z axis of both the aft and the forward dynamometers is along the propeller shaft, hence force in this direction corresponds to the thrust. For the blade dynamometer, the z axis is aligned with the vertical radial direction of the blade it is attached to.

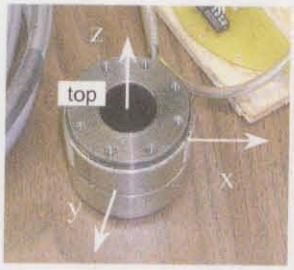


Figure 4. Six component AMTI load cell

The two bearing dynamometers, aft and forward, measure the loads on the drive shaft. The shaft thrust is computed using the information obtained from these two dynamometers. The aft dynamometer is mounted after a thrust de-coupler, therefore, is not expected to encounter as large forces or moments as the other one.

The blade dynamometer is housed inside the hub and attached to one of the blades. A rotary position transducer will record the angular position of this blade. This will enable an accurate matching of the forces and moments due to the contact of the blade with ice.

The wiring for the strain gage and the blade dynamometer is run through the drive shaft and a set of slip rings to a signal conditioner. Due to the high rotational speed and the requirement of a large number of data points over each blade-ice contact, each of these channels will be sampled at 5000 Hz.

3 ICE TANK

3.1 Ice Tank Facilities

The National Research Council of Canada's Institute for Ocean Technologies (IOT) ice tank is 3m deep, 76m long and 12 m wide. A set up area is separated from the ice sheet by a thermal barrier door in order to facilitate setup while the ice sheet is growing. At the opposite end of the tank is located a sloped ramp leading into a melt pit. This pit has an insulated cover allowing an ice sheet to be grown while the remains of the previous one are melting. The ice tank is equipped with a towing carriage that is capable of velocities from 0.1 to 4.0 m/s. It is designed with a central testing area where a test frame, mounted to the carriage, allows the experimental setup to move transversely across the entire width of the tank. The control room is thermally insulated and houses the computer equipment for the drive control and the instrumentation racks for the model test transducers. [4]

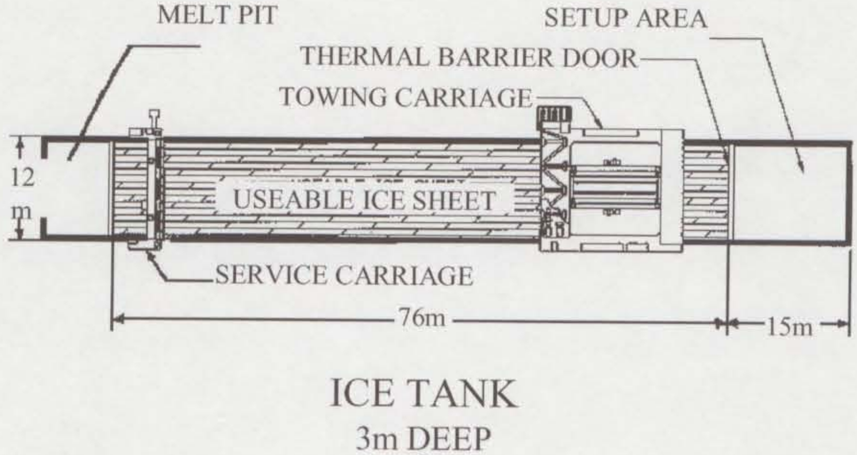


Figure 5. A schematic of the ice tank

For testing in the ice tank, model EG/AD/S ice is used. EG/AD/S ice is designed to provide the scaled flexural failure strength of real sea ice and is made up of a mixture of water, (e)thylene (g)lycol, (a)liphatic (d)etergent and (s)ugar.[5] This formulation has been improved by incorporating tiny air bubbles in the ice to correct its density to give the same buoyancy as is found full-scale The creation of the ice sheet begins with a seeding process and it continues to grow at a temperature of around –20°C until it reaches the target thickness. In order to obtain the target strength the ice is tempered. This involves warming the facility to a temperature of +2°C in order to soften the ice. The tempering process mentioned above causes the disintegration of model ice along the grain boundaries. Due to this, the failure behavior of the model ice is different than that of sea ice. Nevertheless, the failure envelopes of EG/AD/S ice are reported to be as good as the other model ices', even superior in some aspects.

3.2 June 2003 Tests

So far, preliminary ice tank tests have been carried out. The tests consist of open water tests and tests in ice. Two separate ice sheets were used during the ice testing. In total 84 runs were done: 36 in open water and 48 in level ice conditions. During the tests the azimuthing angle was varied from 0^o to 180^o degrees in 45^o intervals with both push and pull mode. Additionally, carriage velocity was varied between 0 and 3 m/s. During the test two propeller rotational speeds were used, 10 rps (revolutions per second) and –10 rps.

The target flexural strength for the tests was 60 kPa at the start of each set of experiments. The ice sheet was generated with a target thickness of 60 mm. During the experiments the density, thickness, flexural, compressive and shear strength values were sampled periodically in order to record the variations in the ice properties. These measurements and tests were conducted as per the standards for ice tank testing.

3.2.1 Flexural strength

The target flexural strength for the start of the experiments was 60kPa. The following figure [Figure 6] shows the flexural strength of both ice sheets used as a function of time.

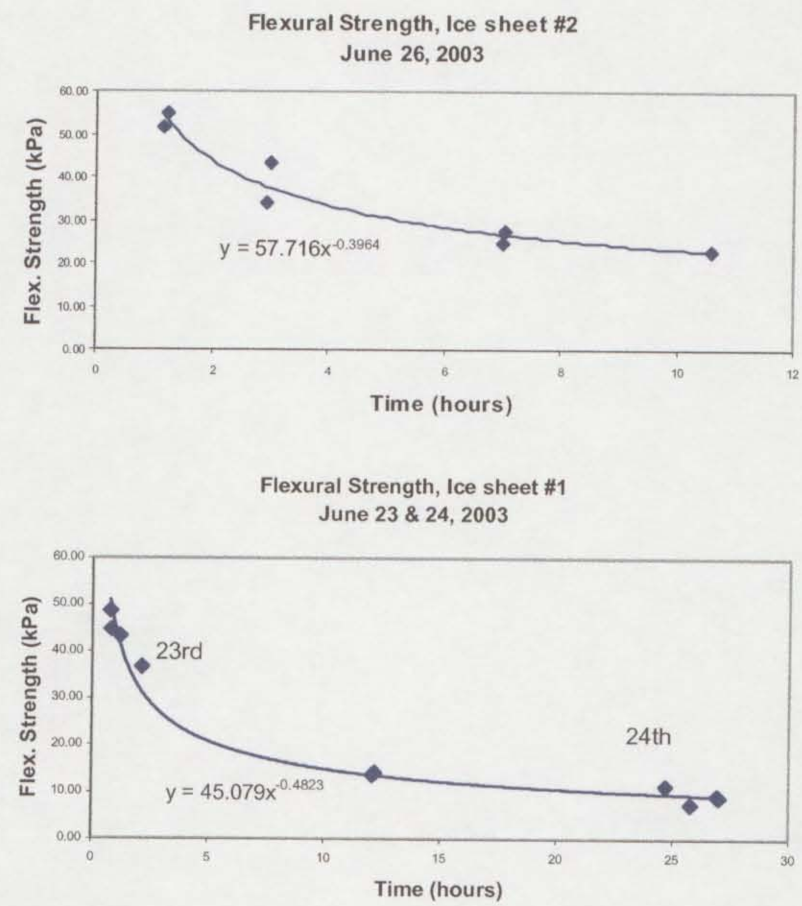
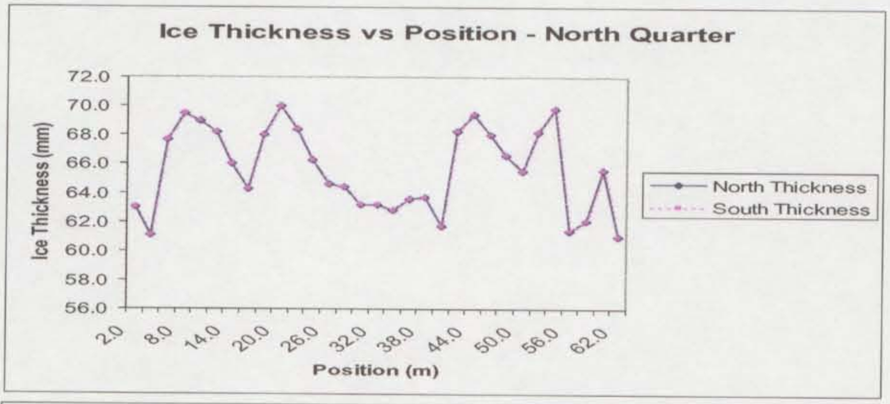


Figure 6. Flexural strength of the ice sheets as a function of hours since tests began.

These graphs show that as time passed the flexural strength of the ice sheet diminished. For the first ice sheet, the flexural strength decreased remarkably. This was due to the fact that it was used over two days. The strength during the tests undergone on the 24th was less than 10kPa compared to the 60kPa goal that was met for the first couple of tests done using this ice sheet. The decrease on the 26th was less drastic, the last tests being done in ice with a strength of around 25kPa. As can be seen in the graph the flexural strength of the ice decreased the most rapidly within the first five hours of testing.

3.2.2 Ice thickness

The objective for the ice thickness was to have a constant thickness of 60mm. During the tests the thickness was measured for the north, south and center sections of the sheet. For each section the measurements were taken every two meters along the length on each side of the section. The following graphs [Figure 7] show the thickness of the external sections as a function the length of the ice sheet.



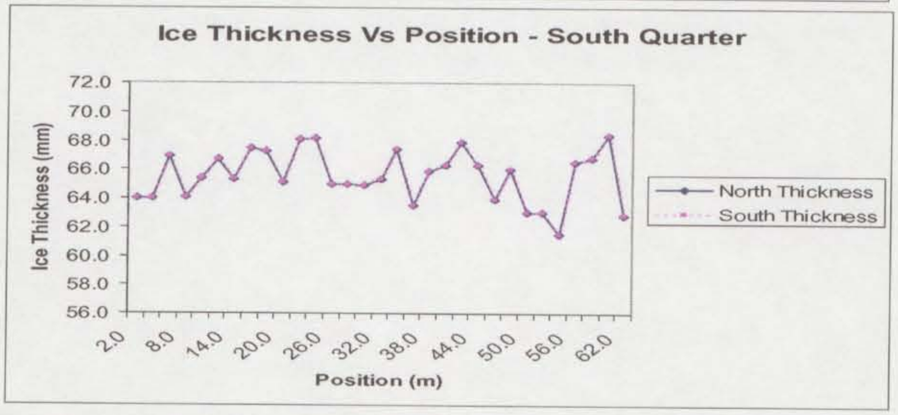


Figure 7. Ice thickness as a function of position for the three sections

For both the South Quarter and the North Quarter the ice thickness measured on the north side and the south side is identical. The Center Profile is gained by taking the south values from the North quarter and the north values for the South quarter. The average thickness for all the graphs is around 65mm.

In plotting the thickness of the ice sheet as a function of time for each ice sheet [Figure 8], as was done for the flexural strength, the changes that are occurring in the ice sheet over time can be seen.

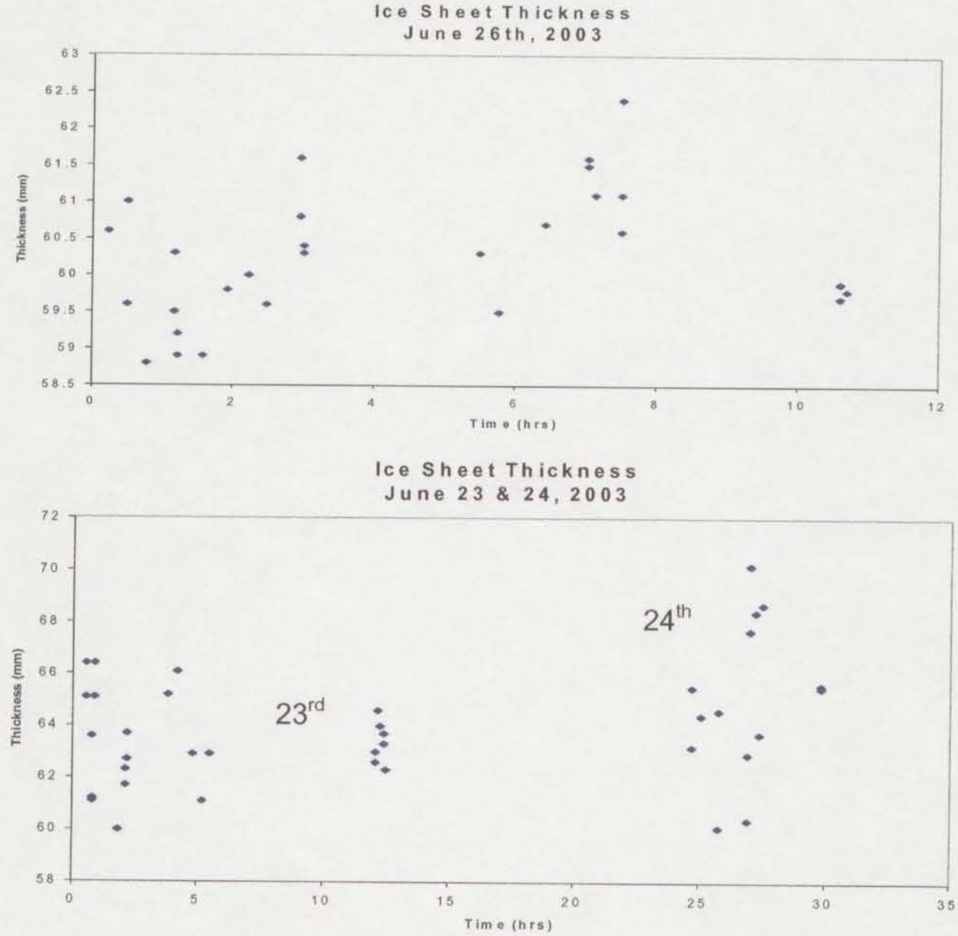


Figure 8. Ice sheet thickness as a function of time

Unlike the flexural strength the ice thickness the ice thickness stays relatively constant as time passes. For the ice sheet used on the 23rd and 24th the thickness varied between 60 mm and 70 mm whereas the ice sheet on the 26th had a thickness between 59 mm and 62 mm.

3.2.3 Compressive strength

Compared to both the flexural strength and the thickness for each ice sheet, the compressive strength was measured less frequently. On the 23rd and 24th of June, the compressive strength of the ice sheet varied between 60kPa and 167kPa. The strength of the ice sheet on the 23rd diminished significantly over time but when tests resumed on the 24th the ice sheet had returned to its original strength around 165kPa. The second ice sheet, used on the 26th of June, shows much less variation in compressive strength. The sheet had an initial strength of around 155kPa and near the end of the day, when it was at it's weakest, it had a strength of 115kPa.

4 ANALYSIS OF THE TARE VALUES

4.1 Shaft Speed Tare Values:

The tare values were taken when the system was at rest with the no shaft rotation, a carriage speed of zero and no ice contact. In the Figure 9 the recorded tare values for the shaft speed are plotted. At a couple of points the shaft speed is not zero as would be expected. It was concluded that the tare values were not properly taken for runs 2 to 7, 31, 32, 54, 56, and 75.

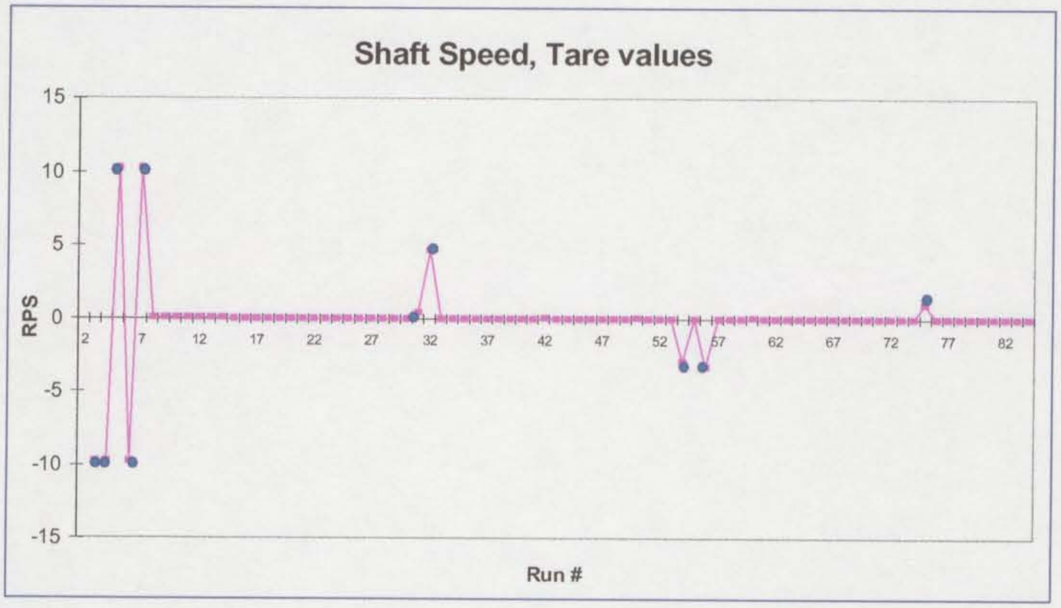


Figure 9. Shaft speed (tare values) versus run number

4.2 Tare Values for the Aft and Forward Load Cells

Analysis of the tare values was done in order to identify any problems that may have occurred during the experiments. The following graphs [Figure 10] show the forces measured by the dyno's while the system was at rest, as a function of the run number. The points that are circled correspond to the runs where the shaft speed tare values were not zero, as discussed earlier.

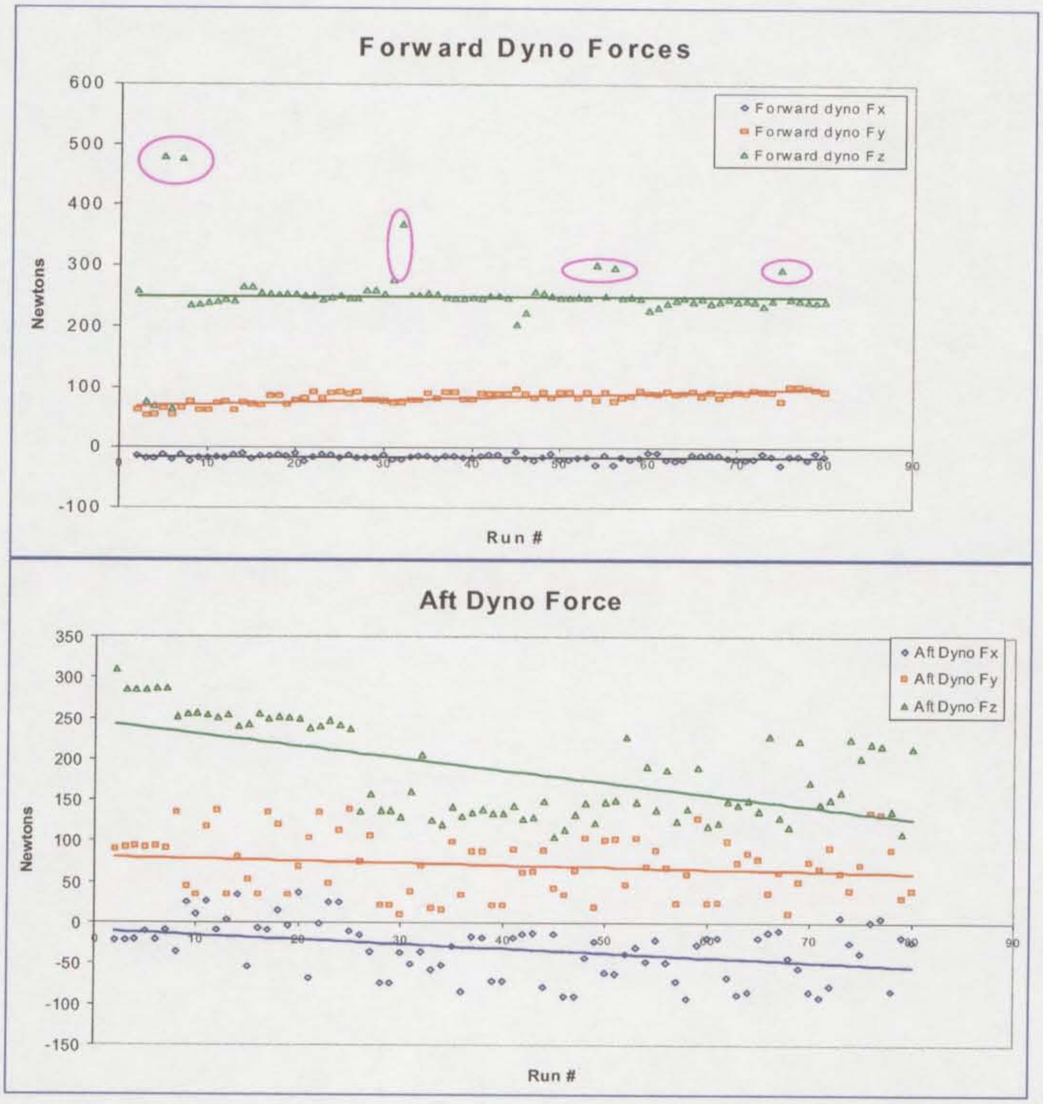


Figure 10. Tare values for the forward and aft load cells versus the run number

These graphs indicate that the output coming from the aft dyno is not very stable. It is important to note that all the forces (Fx, Fy, and Fz) do not all vary in the same fashion. The variation is very large, up to 104 Newtons if you take the maximum and minimum of each force.

In order to determine the cause of the problem with the aft dyno an analysis of the original plans was done. Figure 11 shows the assembly of the aft dyno with the shaft and the thrust decoupler. The problem probably resulted from the propeller shaft pivot engaging the aft dyno in a way it was not supposed to. It was inducing loads inside the bore hole due to the bending of the shaft.

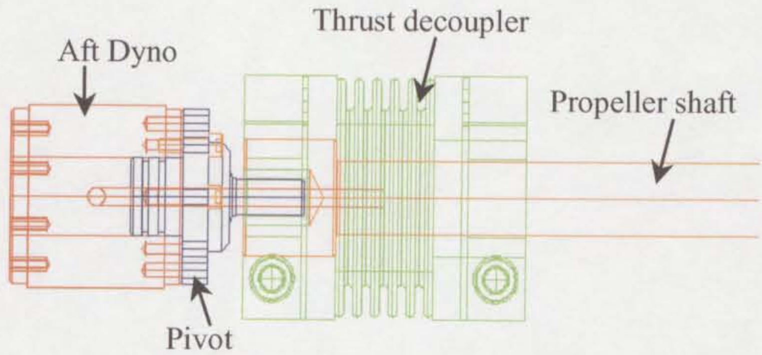
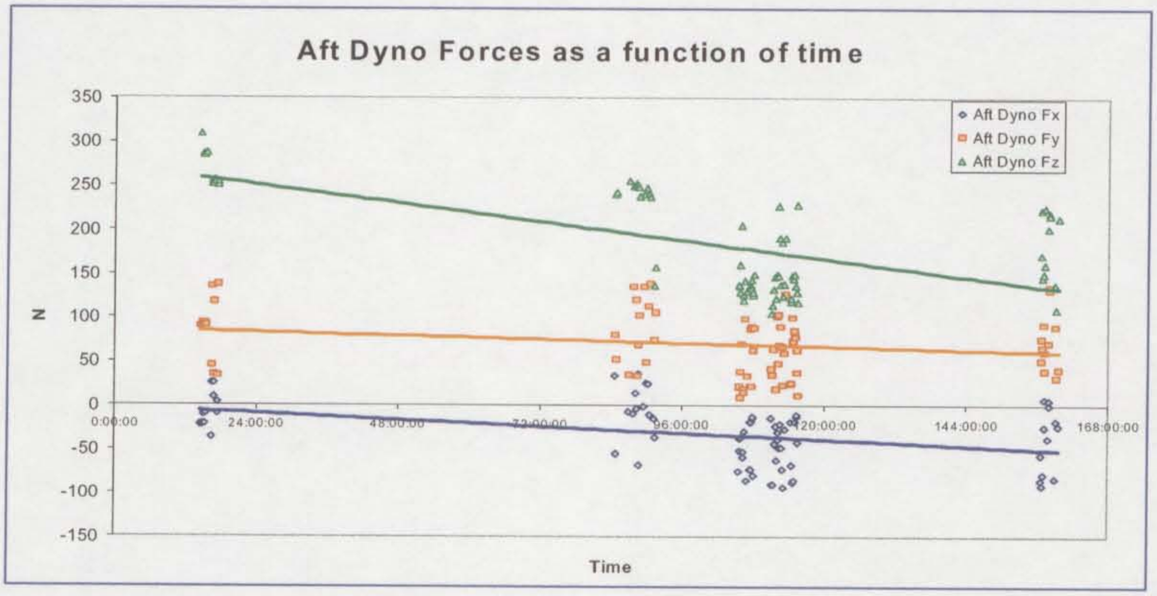


Figure 11. Installation of aft dyno

4.2.1 *Drift in the forward and aft load cells*

The tare values for the forward and aft load cells were analyzed in order to quantify the drift experienced over time. In order to do this it was necessary to approximate each data set by a straight line. This was done by plotting the data as a function of time and setting up a linear approximation. The following [Figure 12] graphs show the linear regression lines for each load cell.



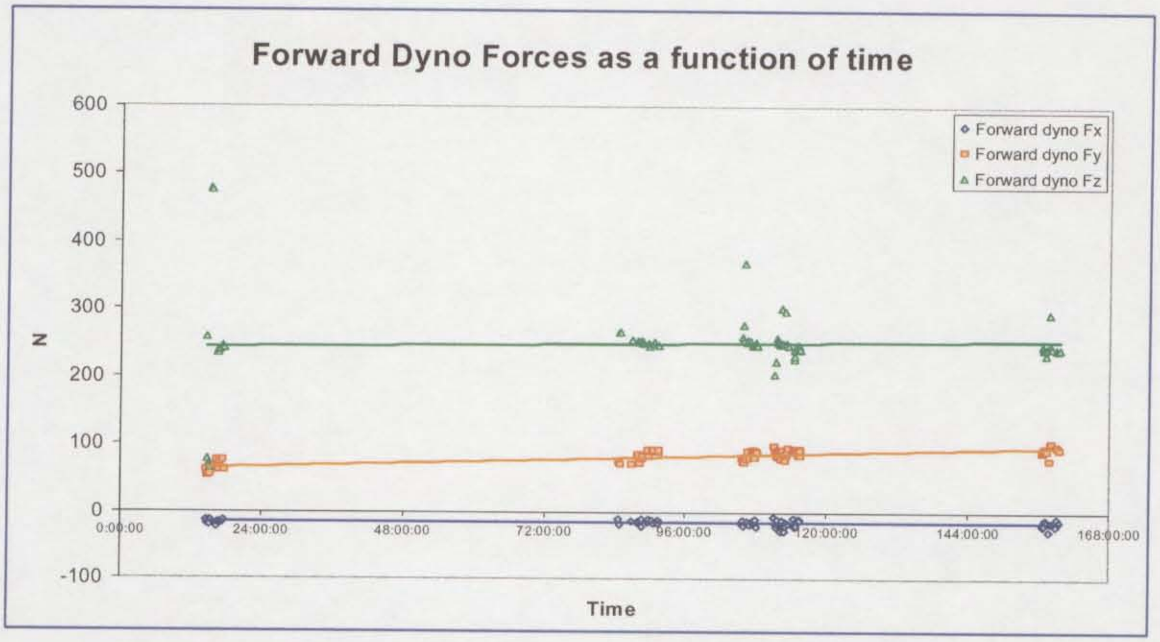


Figure 12. Drift analysis for the aft and forward load cells

To determine the validity of each line the standard deviation on each set of data was also calculated. Tabulated below [**Error! Reference source not found.**] are the standard deviations and the slopes of the linear regressions for each set of data. The smaller, in absolute value, the slope the less drift there is. In general the standard deviations for the aft dyno data are much larger than those of the forward dyno. The two exceptions are highlighted in grey. The standard deviation for Fz on the forward dyno is large because of the points that were affected by the non-zero shaft speeds. The moments in x an y for the aft dyno had particularly large standard deviations.

Table 1. Drift analysis for the forward and aft load cells

	Standard Deviation	Slope of Drift line
Aft Dyno Fx	33.7	-7.4108
Aft Dyno Fy	36.1	-4.0358
Aft Dyno Fz	57.5	-20.957
Aft Dyno Mx	1.9	-26.634
Aft Dyno My	1.5	-11.524
Aft Dyno Mz	0.3	-3.8438
Forward dyno Fx	4.4	0.0411
Forward dyno Fy	11.4	5.3829
Forward dyno Fz	54.2	1.6216
Forward dyno Mx	0.3	6.4033
Forward dyno My	0.4	4.5088
Forward dyno Mz	0.3	5.9611

4.3 Global Load Cell Tare Values

The drift analysis done on the global load cell was less in depth than that done for the forward and aft load cells. This is due to the fact that more information regarding the dynamometers in the global dyno will be gathered during the calibration that is discussed in section 6.2 "Calibration of the Global Loads Dynamometer". In this section the tare values taken during the preliminary test phase will be looked at in order to pick out any anomalies.

The following figure [Figure 13] shows the tare values taken for the global dyno in the x direction. Also present on this graph is a plot of the shaft speed tare values. The graphs of the forces in the y and z directions have similar properties and the following analysis applies to all of them.

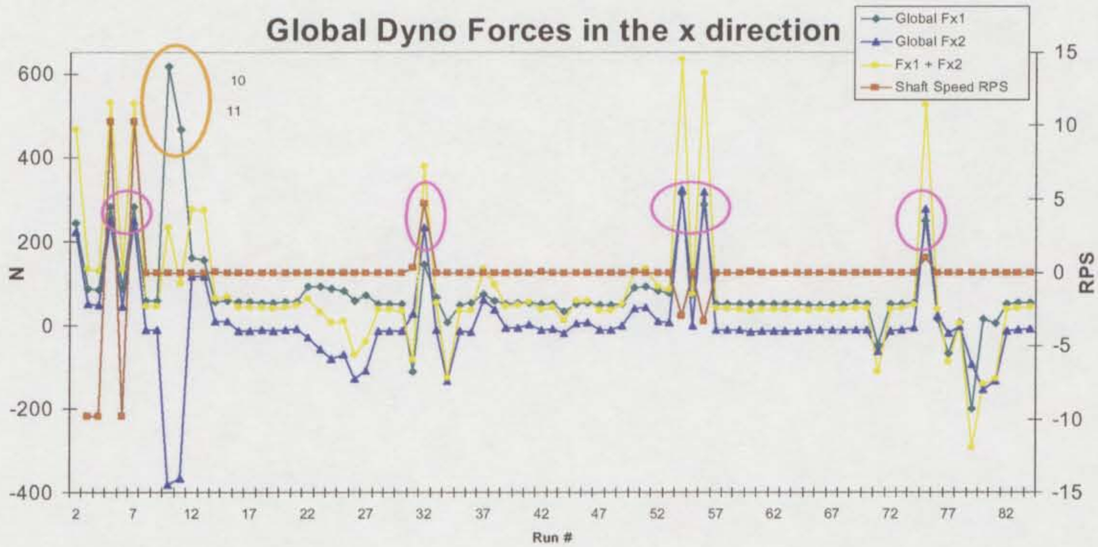


Figure 13. Drift analysis and shaft speed in the x direction, global load cell.

The points that are circled in pink correspond to the runs where the shaft speed for the tare values was not equal to zero. The data collected during runs 10 and 11 (circled in orange) is much different than the data collected during subsequent runs. The forces in all directions seem to either increase or decrease significantly. Unlike those circled in pink, points 10 and 11 do not correspond to the case where the shaft speed was not zero. Figure 14 shows the global dyno forces as well as the azimuthing angle. We can see that for runs 10 and 11 the azimuthing angle was changed from 180° to 135°.

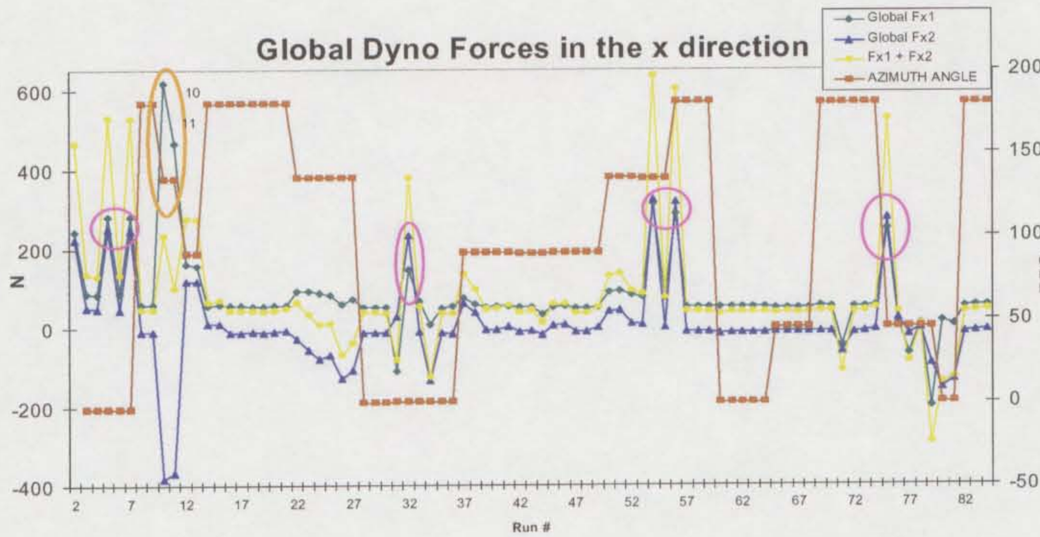


Figure 14. Drift analysis and azimuthing angle in the x direction, global load cell

Figure 15 shows the moments calculated using the global dyno forces. During runs 10 and 11 there were large moments around x and z.

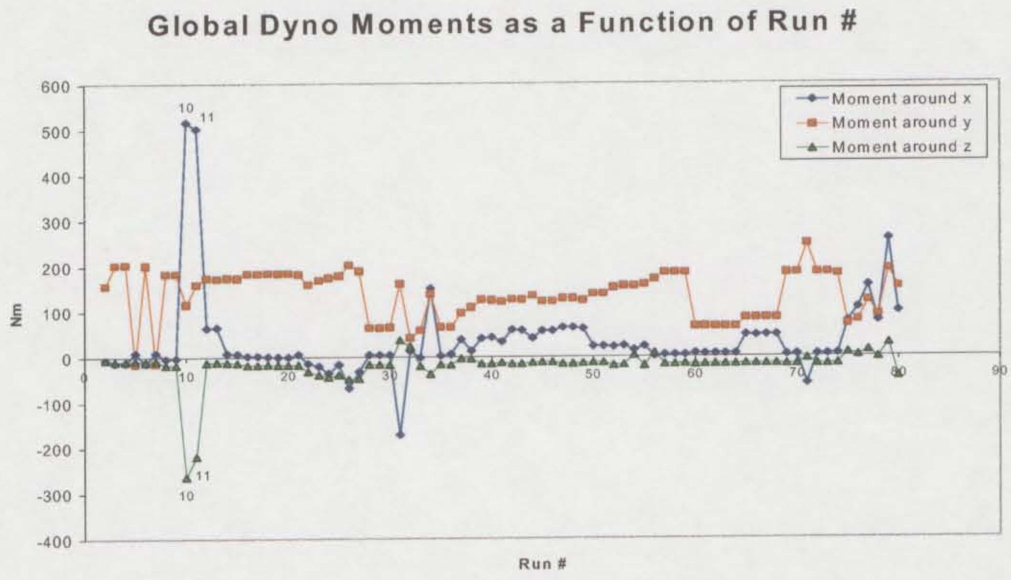


Figure 15. Moments measured by the global load cell

In changing the angle, it is possible that the equipment was not set properly resulting in a force that would not normally be there. During the rest of the angle changes, however, there seems to be little effect on the data. It is clear that these two data points are due to some outside forces or conditions thus they will not be considered in further analysis.

5 RESULTS

In order to determine the amount of force applied on the propeller by the ice itself we looked at the cyclic loading on the dynamometers. One interesting aspect of the analysis was the fact that we were able to determine where each blade was at a given time because we knew the angle of rotation. The rotary position transducer used to measure this was placed on one blade. The angle zero was set to be when the blade was perpendicular to the ice sheet. Using the different cut depths for each run the angle at which the blade hit the ice sheet was found. Figure 16 shows the schematic representation used during this analysis.

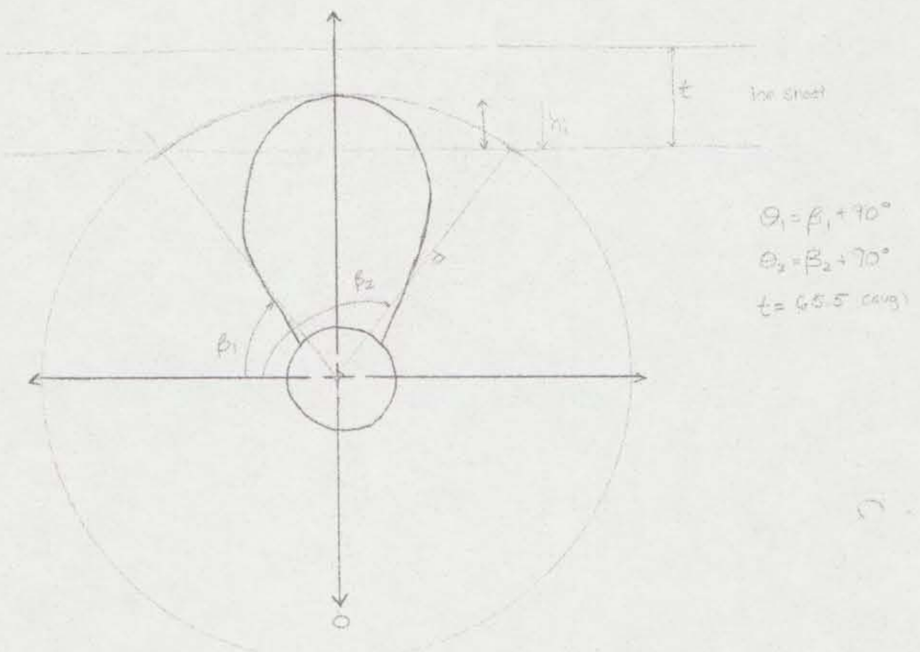


Figure 16. Angle of blade rotation

In order to find the angles during which the blade was in the ice the following assumptions were made.

- In between measurements, the depth of cut did not change.
- The ice sheet remained in the same position during the whole run
- The angle where the blade hit the ice corresponds to the center of the blade.

The following table [**Error! Reference source not found.**] shows the angles calculated for each run.

Table 2. Depth of Cut and Angle of Entry and Exit for Each Run Number

Run numbers	Depth of cut (mm)	θ_1 (degrees)	θ_2 (degrees)
14-21	37	151	208
22-24	39	150	209
25-27	20	159	201
28-44	37	151	208
69-75	10	165	194

76-85	19.5	159	200

5.1 Software Description

In order to determine where the maximum and minimum forces on each of the dynos occurred it was first necessary to compile a Matlab script that would output the information for analysis. This script was named “BladeAngle.m” and can be found in Appendix A.

5.1.1 Software Logic

The program can analysis forces based on either the entire set of data or on a subset of data defined by the user. For time purposes it is recommended that a subset of data be used because of the computational capacities of the computers being used and the large amount of data to be analyzed. First the user selects which run that they wish to analyze and the data for this run is called by Matlab (from a user defined directory specified in the script) and entered into matrix format. From there a graph of shaft speed and carriage velocity as a function of time is displayed to the user so that they can specify which time interval to use. From there the user is prompted to input whether they want to analyze all the data or just the ice contact angles that are specified for each run within the program, and the dyno the wish to analyze.

5.1.2 Program Output

The output of the program varies depending on which dyno the user chooses to analyze. For the global dyno the program will output two different sets of data, one for the measured forces and one for the calculated forces. The measure forces include six forces, two in the x-direction, one in the y-direction and three in the z-direction. For these forces the program outputs the mean values, standard deviation, the maximum and minimum values and the angles associated with each. Also, for the global dyno the program calculates the total forces in all three directions, along with the moments in each direction. To calculate the moments, a sub-program called “Moments” is used which utilizes the measured values along with the moment arms associated with each. For the script of this program see Appendix B. For the calculated values, the program outputs the same data as for the measured values.

For the six component dynometers in the forward and aft positions of the pod, the program outputs some additional information. Along with the mean value, standard deviation, maximum and minimum values and the angles associated with each, the program also outputs the absolute maximum force seen by the dyno, and the maximum change in force from one time step to the next.

Also, with regards to all the dynos, the program creates and displays plots of force vs time. Superimposed on the graph is the theta verses time graph. These plots enable the user to see the direct correlation between the forces measured and the angle of the blade.

5.2 Forward Dynamometer

The forward dyno, located closest to the propeller, measures the loads on the drive shaft. The three forces measured help determine the loads experienced by the shaft at this location as well as the cyclic loading experienced by the bearing. The following figure [Figure 17] outlines the orientation of the x,y,z axes. The z axis points towards the location of the propeller and the shaft runs parallel to it through the center of the load cell.

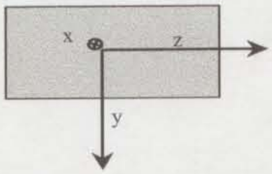


Figure 17. Axis orientation of the forward load cell

5.2.1 Forces in the y direction

In conducting our analysis it was discovered that in some of the runs, number 21 in particular, the values for the forces in the y direction exceeded the maximum load that could be measured by the load cell in certain directions. The result of this was a sinusoidal type output with flat tops and bottoms, the maximums and minimums having been chopped off. Despite the fact that this proves to be problematic when looking at the magnitude of the force, it does make the location of these extremes easier to visualize. The following figure [Figure 18] shows the forces on the forward dyno in the y direction during 0.1 seconds of run 21. The time period chosen was from 40 to 40.1 seconds into the run. This time segment is representative of the data collected during the rest of the run.

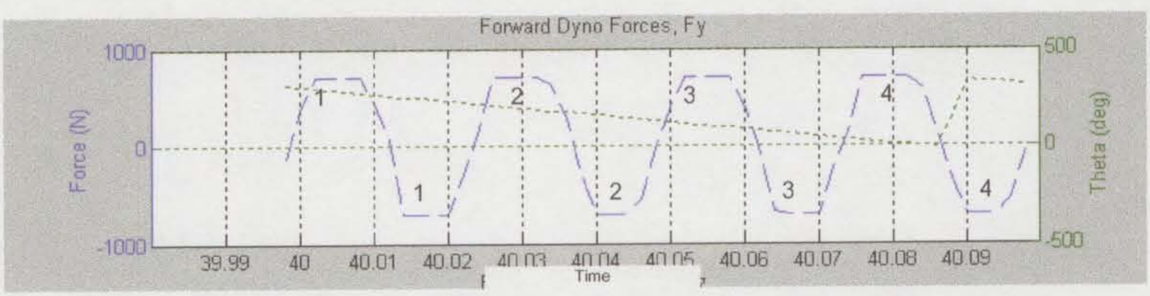


Figure 18. Forward Dyno Forces, Fy versus time for Run 21

In counting the number of complete cycles that take place during one full rotation of the propeller, the speed of rotation of the propeller can be verified. There are four peaks for each full rotation and each full rotation takes about 0.1 seconds. Each peak represents the maximum force on one propeller blade. That means that each blade makes a full rotation in 0.1 seconds. From this, the speed of the propeller is verified as being 10 Hertz, or 10 rotations per second as per the conditions under which run 21 was conducted [**Error! Reference source not found.**].

Table 3. Run 21 testing details

Carriage Speed	Propeller speed	Azimuthing Angle	θ_1 (degrees)	θ_2 (degrees)
0.5 m/s	10 rps	180	151	208

The program "BladeAngle.m" gives the angles at which the maximum force is found. In order to determine where each blade is with respect to the zero axes a correction factor must be used. The four blades are at 90 degrees from each other thus, by adding and subtracting multiples of 90, we can find the position of each blade at a certain time. **Error! Reference source not found.** summarizes the angles for each blade that correspond to the maximum and minimum forces experiences by the forward dyno in the y direction.

Table 4. Locations of the maximum forces

Positive y direction				
Force (N)	**Peak number**	**Interval (degrees)**	**Correction Factor**	**Blade Position interval**
708.3	1	295.4-273.4	-90 degrees	205.4 to183.4
	2	208.7-187.1	--	208.7 to187.1
	3	114.4-92.3	+90 degrees	194.1 to182.3
	4	26.2-4.6	+180 degrees	206.2 to184.3
Negative y direction				
Force (N)	**Valley number**	**Interval (degrees)**	**Correction Factor**	**Blade Position interval**
-707.0	1	244.2-230.1	-90 degrees	154.5 to140.1
	2	157.8-143.2	--	157.8 to143.2
	3	62.9-48.4	+90 degrees	152.9 to138.4
	4	334.4-324.0	-180 degrees	154.4 to144.0

The following two sketches help to visualize this information. The maximum force magnitudes occur as the blades of the propeller enter and exit the ice sheet.

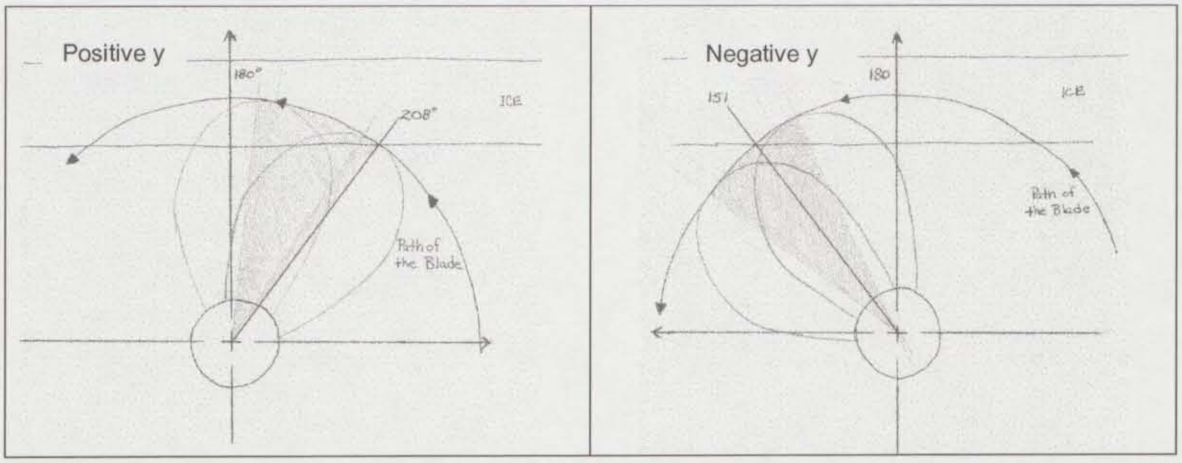


Figure 19. Sketch of the location of the maximum and minimum forces, forward dyno, Fy.

As the blade enters the ice and as it exits larger forces are put on the forward dyno than when it is vertical in the ice. In doing the same analysis for runs 20 and 22 we find the same results. Run 19 is very similar, the only difference being that the blade is turning in the opposite direction. Runs 25 and 26 do not reach the maximum load of the load cell as often but the location of the extremes, both positive and negative, show that the magnitude of the force is located as the blades enter or leave the ice sheet. All of these runs have an azimuthing angle of 180 degrees

Table 5. Summary table for runs 19 to 21

Run #	Force (N)	Average Blade Position Interval	Position
19	708.3	177.9 to 195.5 degrees	Exiting Ice
	-707.0	138.0 to 147.1 degrees	Entering Ice
20	708.3	201.5 to 187.3 degrees	Entering Ice
21	708.3	203.6 to 184.3 degrees	Entering Ice
	-707.0	154.9 to 141.4 degrees	Exiting Ice

In analyzing other runs that had different azimuthing angles it was found that the maximum and minimum forces occurred in the same locations as those runs just mentioned. In the following figures we see two different runs with different azimuthing angles. These runs both have a carriage velocity of 0.5 m/s and a shaft speed of 10 rps. In run 43 (tractor mode), Figure 20, we can see that the angles where the minimum forces are located correspond to the angle θ_1, which, in this case, represents the blades leaving the ice. In run 73 (pusher mode), Figure 21, the maximum forces are located around θ_1 or as the blades are exiting the ice.

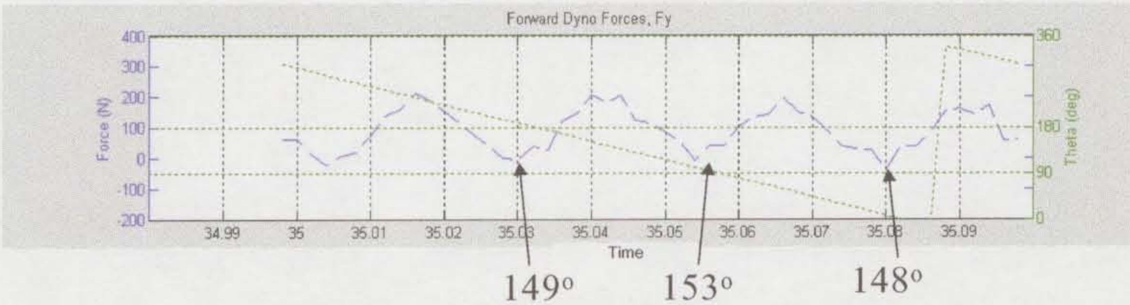


Figure 20. Run 43, AZ angle 90°, θ_1=151°

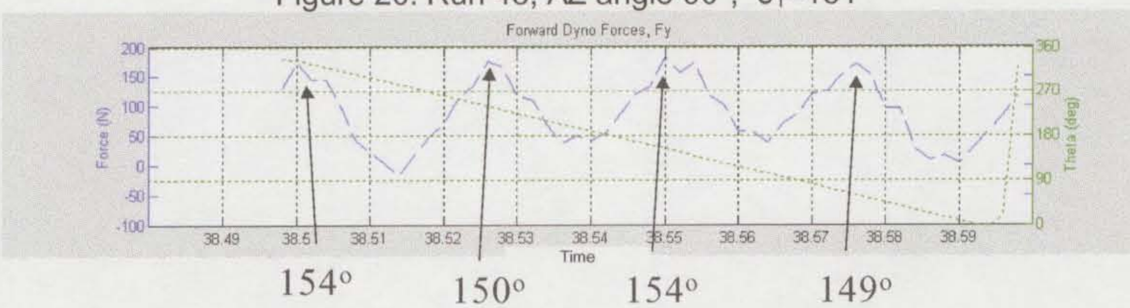
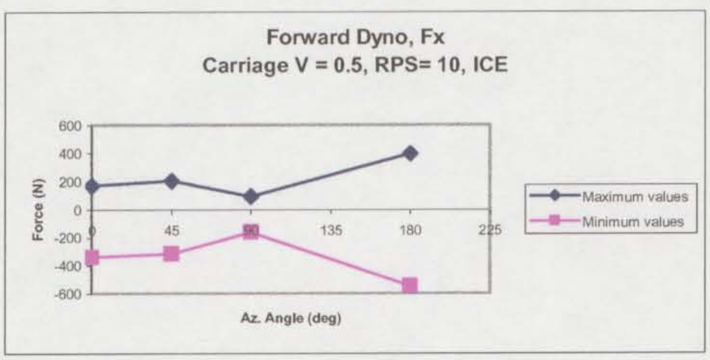


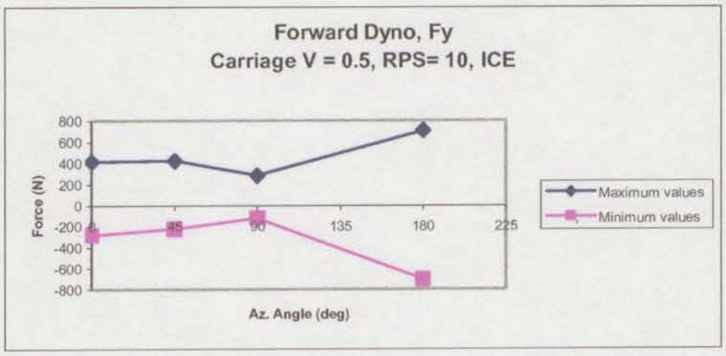
Figure 21. Run 43, AZ Angle of 45°, θ_1= 159°

5.3 Effects of azimuthing angle

5.3.1 Forward Dynamometer

The preliminary test included runs being done with varying azimuthing angles. The angles tested were 0°, 45°, 90°, 135° and 180°. By looking at the maximum and minimum forces experienced by the forward load cell when the azimuthing angle was varied while keeping the carriage velocity and rotational speed constant, it is possible to determine at which azimuthing angle the cyclic loading is greatest. The following figure shows the maximum and minimum forces in each direction as a function of the azimuthing angle chosen.





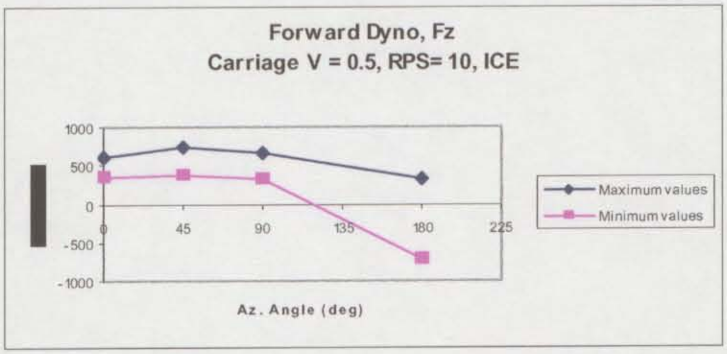


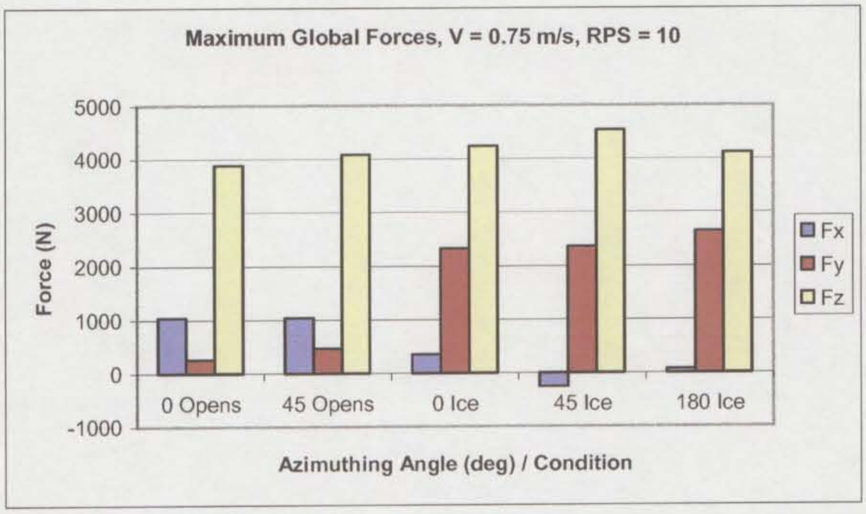
Figure 22. Force versus az. angle for runs with a carriage velocity of 0.5m/s and 10 rps propeller speed

In both the x and y directions the largest maximum and minimum forces occur when the azimuthing angle is 180°.

5.3.2 Global Load Cell

The global loads dynometer is positioned at the end of the strut farthest away from the pod and measures the global forces experiences by the entire system. This loading represents the forces that would be experienced at the stern of the vessel where the propeller would be attached.

As previously demonstrated with the forward dyno, looking at the maximum and minimum forces experienced by the global dyno when the azimuthing angle was varied, while keeping the carriage velocity and rotational speed constant, it is possible to determine at which azimuthing angle the cyclic loading is greatest. The following figure shows the maximum and minimum forces in each direction as a function of the azimuthing angle. The azimuthing angles chosen in this comparison were 0, 45 and 180 degrees. Also, the comparison of maximum and minimum forces was also completed for two different carriage velocities, 0.3 m/s and 0.75 m/s.



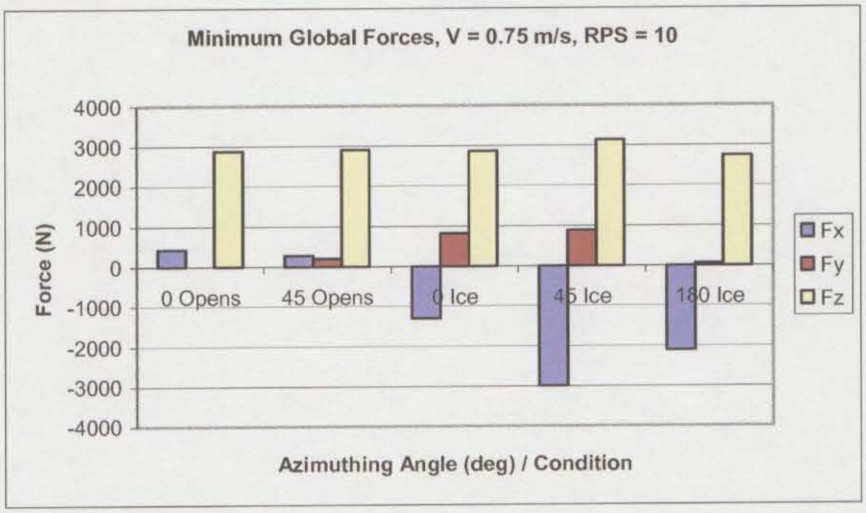
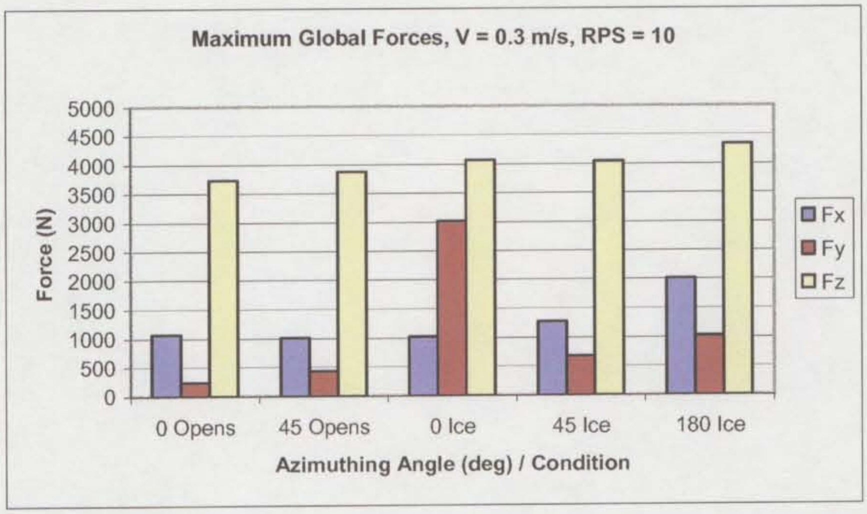


Figure 23: Maximum and Minimum Forces vs Azimuthing Angle for the Global Dyno (V=0.75 m/s)

The next set of graphs illustrate the same points as the previous except for the fact that they data used consists of a carriage velocity equal to 0.3 m/s.



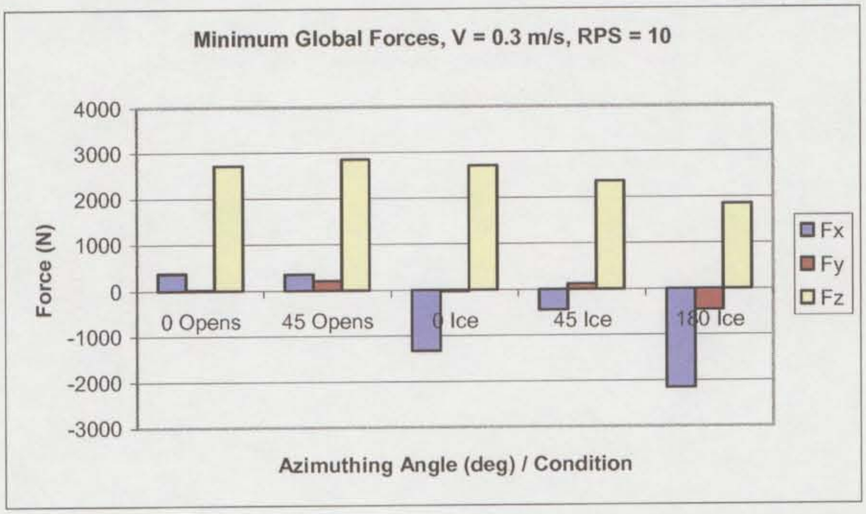


Figure 24: Maximum and Minimum Forces vs Azimuthing Angle for the Global Dyno (V = 0.3 m/s)

No conclusions regarding the effect of the change in azimuthing angle have been draw as, at the time this report was written, the analysis had not been completed.

6 PREPARATION FOR FUTURE TESTING

The data analysis presented in this report was for the preliminary tests on the performance of podded propellers in ice. The goal of these tests was to determine any problems with the experimental apparatus and/or the test procedure. If any problems were encountered they could be remedied before the final testing was to take place. This testing is tentatively scheduled from the end of February or early March 2004. Some preparation has been completed for this testing. Some of the preparation for this set of tests has included and will include temperature sensitivity analysis for the AMTI six-component load cells and the calibration of the global load dynometer.

6.1 Temperature Sensitivity of AMTI Six-Component Load Cell

To determine the effect that temperature will have on the use and calibration of the AMTI load cells a series of tests were performed. The tests involved a series of pushes and pulls applied to the load cell by means of a specially fabricated jig and electric motor arrangement. This test was performed at three different temperatures. The temperatures that were chosen were room temperature (approximately 21 degrees Celsius), 5 and –2 degrees Celsius. The characteristic graphs for both the aft and forward dynos for each or the stated temperatures are shown below [Figure 25, Figure 26].

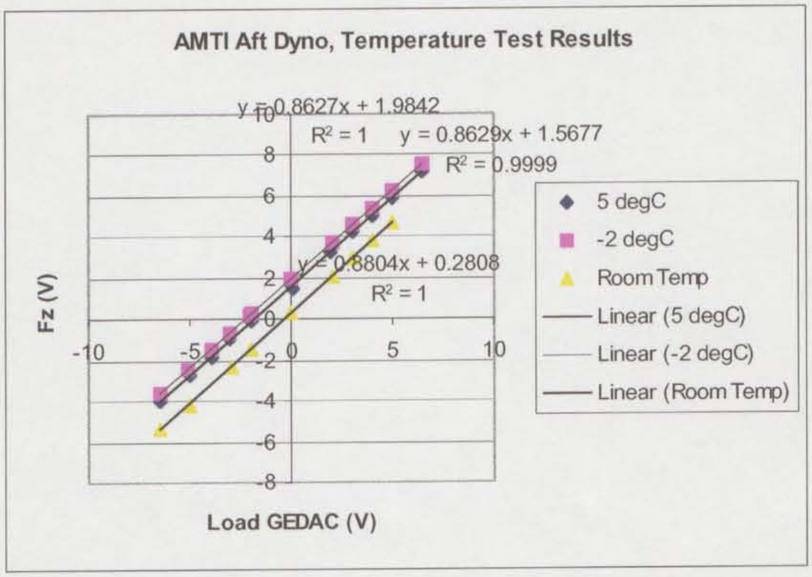


Figure 25: Effect of Temperature on AMTI Aft Dyno Performance

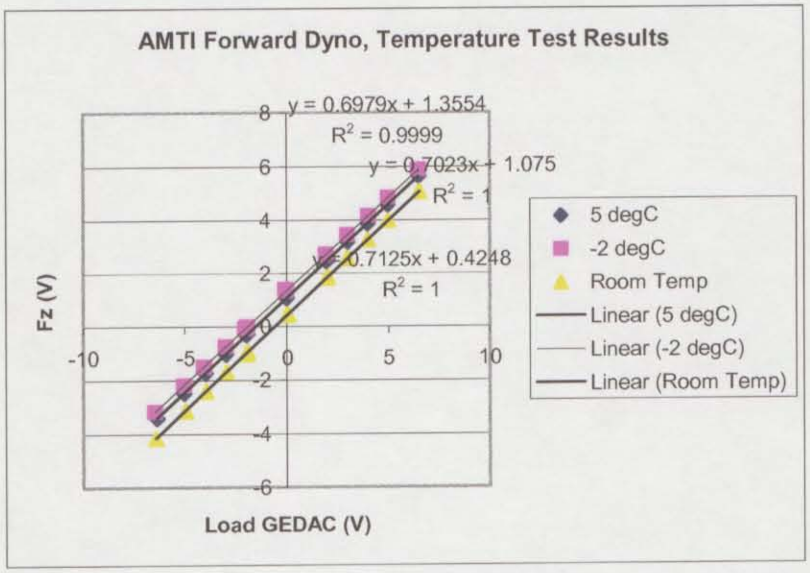


Figure 26: Effect of Temperature on AMTI Forward Dyno Performance

In comparing the results for each AMTI load cell it can be seen that the temperature has very little effect on the slope of the force versus applied load graph. However, the intercept of the line does change considerably at varying temperatures. This is the result that was expected before the test took place and presents a problem that is very easy to solve. For the calibration and use of each of these load cells, the lines must be forced to zero and subsequent readings from the load cell adjusted accordingly.

However, to show the validity of this set of tests, the room temperature tests were conducted multiple times to make sure the acquired data was consistent. Below are the plots for the aft and forward dynos at room temperature for three different trials [Figure 27, Figure 28].

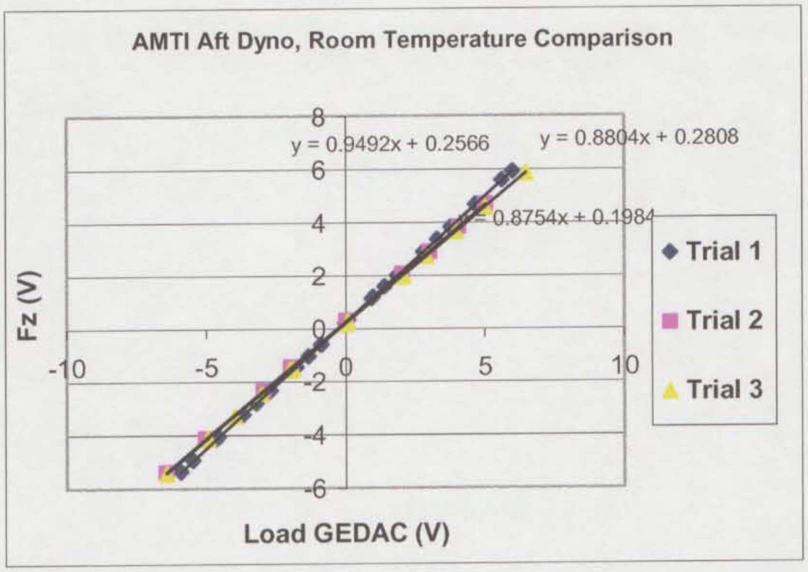


Figure 27: AMTI Aft Dyno Room Temperature Verification

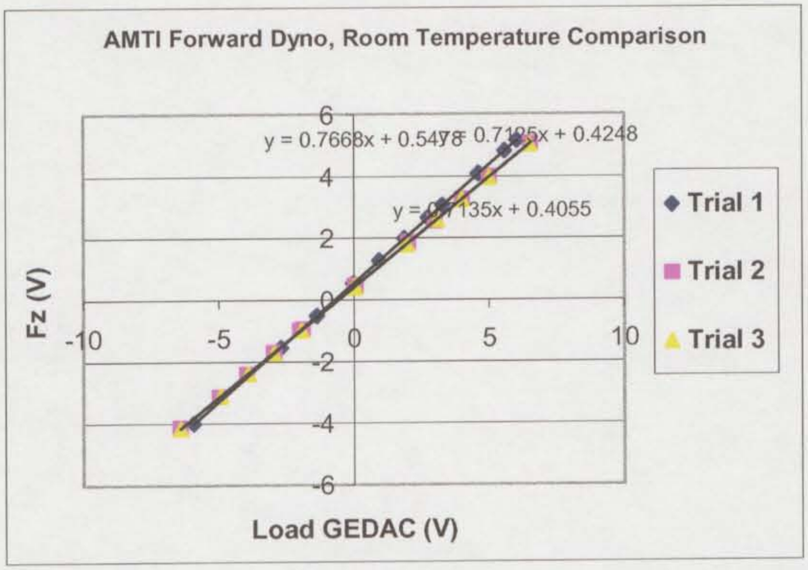


Figure 28: AMTI Forward Dyno Room Temperature Verification

As shown in the graphs the consistency of the tests at room temperature was very adequate with very little variation in slope or intercepts between each trial.

6.2 Calibration of the Global Loads Dynamometer

A comprehensive calibration of the global loads dynometer is required in order to begin the next phase of testing. At the present time only the setup for the calibration has begun. The calibration will involve 18 different setups. The apparatus used to calibrate this global loads dyno is shown in the figure below. Figure 29 is a schematic diagram while of the apparatus that will be used.

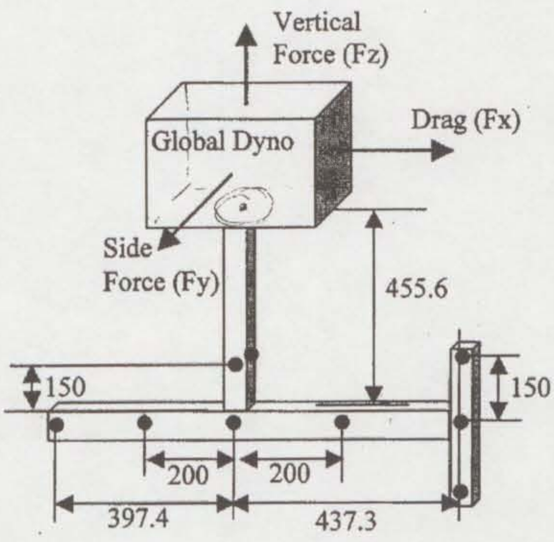


Figure 29: Schematic Diagram for Global Dyno Calibrations

Each of the points on the diagram indicates the place of force application on the calibration apparatus. The load will be applied along all three axes using two hydraulic rams for the X and Y directions and a hanging weight for the Z direction. Different combinations of loading arrangements will be used in each of the 16 setups. The following picture illustrates one of the setups that will be utilized during the calibration sequence.

Figure 30. Setup Used During the Global Dyno Calibration

6.2.1 Calibration Matrix for Global Dyno.

The calibration matrix for the Global Dyno will be determined using a linear curve fit on the calibration points. The program written to do this is called "Calib.m" and runs in *Matlab*. Calib.m reads the files from the test and performs the following analysis. Please see appendix C for the complete program. It reads the file containing the information on the load applied in x, y and z and stores it as matrix P. It then reads the file containing the location of the applied forces and stores it as matrix c.

$$P = [\,Px, Py, Pz]$$

$$c = \begin{bmatrix} \text{(point of application of Px)} \\ \text{(point of application of Py)} \\ \text{(point of application of Pz)} \end{bmatrix} = \begin{bmatrix} x1 & y1 & z1 \\ x2 & y2 & z2 \\ x3 & y3 & z3 \end{bmatrix}$$

Equation 1. Input for Calib.m

The next step is to read the files containing the output coming from the load cells. This is called matrix "MtxR" and it contains a voltage output for each of the six load cells.

$$MtxR = [Rx1, Rx2, Ry, Rz1, Rz2, Rz3]$$

Equation 2. Load cell output used in Calib.m

Using the matrix "P", containing the loads applied in each direction, and matrix "c", containing the points of application of each force, the function "ForceFinder.m" (appendix

D) calculates the force experienced by each load cell during the application of the loads in question. These forces are stored as the matrix called "MtxF".

The equation that ties the applied load, the calibration matrix and the output from the load cells is as follows:

MtxR = MtxCalib*MtxF

Equation 3. Relationship between applied force, output and calibration matrix.

The calibration matrix, "MtxCalib" can thus be found using the following equation:

MtxCalib=MtxR*inv(MtxF)

Equation 4. Calculation of the calibration matrix

6.2.1.1 Functions used in Calib.m

The function "ForceFinder.m" (appendix D) was created to find the force that would be applied on any one of the individual dynamometers on the global dynamometer when a load is applied to the system as a whole. The method used is that set out in the paper *Dynamometer Statics* by W. Raman-Nair and S.N.Chin. In appendix E the set up of the global dyno. is outlined. In order to discover the relationship between the applied load and the reaction forces on the individual dynamometer the following equations were determined with the aid of the set up diagram. In the following equations F_m is the force acting on point m whose position vector relative to 0 is r_m [6].

Force $\mathbf{F}_m$	Position vector $\mathbf{r}_m$
$\mathbf{F}_{z1}=F_{z1}\mathbf{k}$	$\mathbf{r}_{z1}= - x_{z1}\mathbf{i} - y_{z1}\mathbf{j} + z_{z1}\mathbf{k}$
$\mathbf{F}_{z2}=F_{z2}\mathbf{k}$	$\mathbf{r}_{z2} = x_{z2}\mathbf{i} + z_{z2}\mathbf{k}$
$\mathbf{F}_{z3}=F_{z3}\mathbf{k}$	$\mathbf{r}_{z3} = - x_{z3}\mathbf{i} + y_{z3}\mathbf{j} + z_{z3}\mathbf{k}$
$\mathbf{F}_{y}=F_{y}\mathbf{j}$	$\mathbf{r}_{y} = - y_{y}\mathbf{j} + z_{y}\mathbf{k}$
$\mathbf{F}_{x1}=F_{x1}\mathbf{i}$	$\mathbf{r}_{x1} = x_{x1}\mathbf{i} - y_{x1}\mathbf{j} + z_{x1}\mathbf{k}$
$\mathbf{F}_{x2}=F_{x2}\mathbf{i}$	$\mathbf{r}_{x2} = x_{x2}\mathbf{i} + y_{x2}\mathbf{j} + z_{x2}\mathbf{k}$

Equation 5. Position vector and force vector for global dyno.

Seeing as the system is in static equilibrium the following equations can be used [6] with the load applied being **R**= (Rx, Ry, Rz) and the point where the load is applied being **C** = (x_c, y_c, z_c)

$$\Sigma\mathbf{F}_m + \mathbf{R} = 0 \rightarrow \Sigma\mathbf{F}_m = -\mathbf{R}$$
$$\Sigma\mathbf{r}_m\times\mathbf{F}_m + \mathbf{C}\times\mathbf{R} = 0 \rightarrow \Sigma\mathbf{r}_m\times\mathbf{F}_m = -\mathbf{C}\times\mathbf{R}$$

Equation 6. Static equilibrium equations

With these equations a system of equations is developed that has, in our case, six equations and six unknowns. It is this system of equations that "ForceFinder.m" solves to find the vector $\mathbf{F}_m$. For the development of these equations see appendix E-2.

The function "Moments.m" (appendix B) finds the moments in x, y and z when a vector containing the forces experienced by each individual dyno is entered. The equations are straight forward and are outlined in on page 2 of appendix B.

6.2.2 Resultant Applied Calibration Load and Line of Action

The loads applied during the calibration process for the global dynamometer can be replaced by a single load, or resultant load, acting at some point. [6] The program "Resultant.m" (appendix F) was designed to locate the point where the resultant vector is applied by defining its position vector, **OC**. The program is currently set up to generate a three dimensional graph showing the position vectors for a number of different load combinations. The necessary input is a matrix containing the different force combinations in x, y and z. In the same style as described on page 27, the vectors containing the points of application of each force must be entered as well. These are stored in a three dimensional matrix for convenience. Using the function "ForceFinder.m", the matrix F containing the forces on each individual dyno is found for each loading setup. As described in the paper *Dynamometer Statics* [6] the next step includes finding the resultant force vector, forces in each direction, and the vector for the moments around the origin caused by the forces sing the following equations.

$$\mathbf{R} = \Sigma\, \mathbf{F}_m = (F_1+F_2,\ \ F_3,\ \ F_4+F_5+F_6)$$
$$\mathbf{M}_o = \Sigma \mathbf{r}_m \times \mathbf{F}_m = (-F_4{*}Y_{z1}+F_6{*}Y_{z3}-F_3{*}Z_y,\ \ F_4{*}X_{z1}-F_5{*}X_{z2}+F_6{*}X_{z3}+F_1{*}Z_{x1}+F_2{*}Z_{x1},\ \ F_1{*}Y_{x1}-F_2{*}Y_{x1})$$

Equation 7. Evaluate the resultant force vector and the moment vector

Following the steps outlined by W. Raman-Nair and S.N.Chin [6] the position vectors **u**, **v**, **w**. The constant β is found using Equation 8.

$$\beta = (\mathbf{M}_o \bullet \mathbf{v}) / \mathrm{mag}(\mathbf{R})$$

Equation 8. Calculating β

With Equation 9 we have our position vector **OC** as a function of a constant "a" and we can plot the line.

$$\mathbf{OC} = a\,\mathbf{u} + \beta\,\mathbf{w}$$

Equation 9. Position vector for the resultant force

The program "Resultant.m" can perform this analysis for multiple setups and plot them all on the same graph in order to determine whether the calibration setups chosen represent the whole spectrum of possible forces.

7 CONCLUSION

In June of 2003 preliminary tests were done on the podded propellers in ice project. The analysis done helped to identify and fine-tune some of the problems associated with the experimentation procedure and with the equipment itself. During the analysis of the tare data taken during the preliminary tests it was discovered that the aft dyno generated a lot of scatter. This is probably due to a slight bend in the propeller shaft causing the pivot to engage the dyno and resulting in supplemental loads being measured. In looking at the tare values for the shaft speed it was discovered that during a number of runs the tare values were not properly taken resulting the data from these runs being inaccurate. After looking at the tare values taken during the testing an analysis was done to determine at what instances and relative positions the apparatus experiences its maximum loading. For the forward dyno the maximum forces in the y direction (vertical when looking at the assembled structure) occurred when the blades were entering and exiting the ice sheet.

In preparation for the second round of testing to be conducted in February or March of 2004 elements of the experimental setup were tested and calibrated. Testing was done on the two AMTI six component load cells used as the forward and aft dyno's in three different temperature, including that at which the tests will be conducted. These tests showed that the load cells are sufficiently thermally stable to be used during the subsequent round of testing. Another step that had to be taken before the next tests was the calibration of the global load cell. At the time that this report was written this calibration was not completed but was well underway. Sixteen different loading arrangements will be done to complete the calibration. The load will be applied along all three axes using two hydraulic rams for the X and Y directions and a hanging weight for the Z direction.

8 REFERENCES

1. Azipods for last two Carnival liners from Masa-Yards, Naval Architect, November 1995, p. 605.

2. Hamlet joins the Helsingor-Helsingborg shuttle, Naval Architect, July/August 1997, p. 20.

3. Twin-propeller pod with fins from Schottel, Naval Architect, July/August 1997, p. 31

4. Jones, S. J., 1987, "Ice Tank Test Procedures at the Institute for Marine Dynamics," Institute for Marine Dynamics Report No. LM-AVR-20

5. Timco, G.W., 1986,"EG/AD/S: A New Type of Model Ice for Refrigerated Towing Tanks," Conld Regions Science and Technology, Vol. 12, pp. 175-195

6. Raman-Nair, W., Chin, S.N., 2003, "Dynamometer Statics," Institute for Ocean Technologies Report.

Appendix A

```
clear all
close all

%COLUMN DESCRIPTIONS----------------------------------------------------------

disp('MATRIX 1 - channels 1-29')
disp('Column 1 - Time                Column 16 - Forward Fx')
disp('Column 2 - Shaft Speed (RPS)   Column 17 - Forward Fy')
disp('Column 3 - Azimuth Angle       Column 18 - Forward Fz')
disp('Column 4 - Global Fx1          Column 19 - Forward Mx')
disp('Column 5 - Global Fx2          Column 20 - Forward My')
disp('Column 6 - Global Fy           Column 21 - Forward Mz')
disp('Column 7 - Global Fz1          Column 22 - Shaft Torque')
disp('Column 8 - Global Fz2          Column 23 - Input Signal')
disp('Column 9  - Global Fz3         Column 24 - Input Signal')
disp('Column 10 - Aft Fx             Column 25 - Input Signal')
disp('Column 11 - Aft Fy             Column 26 - Input Signal')
disp('Column 12 - Aft Fz             Column 27 - Input Signal')
disp('Column 13 - Aft Mx             Column 28 - Input Signal')
disp('Column 14 - Aft My             Column 29 - Blade Angle')
disp('Column 15 - Aft Mz             Column 30 - Motor Current')

disp('MATRIX 2 - channels 30-34')
disp('Column 1 - Time')
disp('Column 2 - XPull')
disp('Column 3 - YPull')
disp('Column 4 - Test Frame Height')
disp('Column 5 - Carriage Position')
disp('Column 6 - Carriage Velocity')

%INPUT PARAMETERS-------------------------------------------------------------

source_directory = 'C:\Documents and Settings\DuffyD\My Documents\Output\';
source_file_common_part = 'matrix_run_';
source_file_common_part2 = 'matrix_run2_';

%USER INPUTS A FILE NUMBER TO BE ANALYSED------------------------------------

RunNumber = input('Enter the number of the run to be analyzed: ');

%WHILE LOOP USED TO MAKE SURE THE RUN NEMBER IS VALID---------------------

while (RunNumber < 1) | (RunNumber > 84)

   disp('The run number you entered is invalid. It must be between 1 and 84.');
   disp('Please try again');

   RunNumber = input('Enter the number of the run to be analyzed: ');

end

%COMPLETE/FORMAT FILE INPUT STRING----------------------------------------

i = RunNumber;
fname2 = num2str(i);
fname3 = '.DAT';
```

```
if (i < 10 )

   fname1 = '00';
   fname_source  = strcat(source_directory, source_file_common_part, fname1, fname2, fname3);
   fname_source2 = strcat(source_directory, source_file_common_part2, fname1, fname2, fname3);

elseif (i <= 99)

   fname1 = '0';
   fname_source  = strcat(source_directory, source_file_common_part, fname1, fname2, fname3);
   fname_source2 = strcat(source_directory, source_file_common_part2, fname1, fname2, fname3);

end

%LOAD THE TWO MATRICES INTO THE PROGRAM-------------------------------------

matrix = load(fname_source);        %Matrix for channels 1-29
matrix2 = load(fname_source2);      %Matrix for channels 30-34

%USER DECIDES WHETHER TO USE ALL ANGLES OR ICE CONTACT ANGLES--------------

disp('Input whether to use all angles or only ice contact angles');
angle_input = input('1 = all angles, 2 = ice contact angles: ');

if (angle_input == 1)

   theta = fname_source(:,29);

elseif (angle_input == 2)

   %DEFINE THE ANGLES FOR WHICH THE BLADE IS IN CONTACT WITH THE ICE SHEET----

   if (i >= 14) & (i <= 21)
      theta1 = 151.25;
      theta2 = 208.75;
   elseif (i >= 22) & (i <= 24)
      theta1 = 150.46;
      theta2 = 209.54;
   elseif (i >= 25) & (i <= 27)
      theta1 = 158.96;
      theta2 = 201.04;
   elseif (i >= 28) & (i <= 44)
      theta1 = 151.25;
      theta2 = 208.75;
   elseif (i >= 69) & (i <= 75)
      theta1 = 165.16;
      theta2 = 194.84;
   elseif (i >= 76) & (i <= 85)
      theta1 = 159.23;
      theta2 = 200.77;
   end

end

%PLOT GRAPHS OF TIME VS CARRIAGE VELOCITY AND SHAFT SPEED------------------

[m,n] = size(matrix);
```

```
Time = matrix(1:m,1);

ShaftSpeed = matrix(1:m,2);
[m,n] = size(matrix2);

Time2 = matrix2(1:m,1);
CarriageVel = matrix2(1:m,6);

subplot(2,1,1); plot(Time,ShaftSpeed);
title('Shaft Speed vs Time');
xlabel('Time (s)');
ylabel('Shaft Speed (RPS)');
grid on;

subplot(2,1,2); haxis = plot(Time2,CarriageVel);
title('Carriage Velocity vs Time');
xlabel('Time (s)');
ylabel('Carriage Velocity (m/s)');
grid on;

%USER DEFINES WHAT TIME RANGE TO BE ANALYZED------------------------------

BeginTime = input('Enter the start time (in seconds): ');
EndTime = input('Enter the end time (in seconds): ');

%WHILE LOOP TO VALIDATE GIVEN TIME FRAME---------------------------------

while (EndTime <= BeginTime)

   disp('End time must be greater than start time');
   disp('Please re-enter start time and end time');

   BeginTime = input('Enter the start time (in seconds): ');
   EndTime = input('Enter the end time (in seconds): ');

end

FirstRowNumber = BeginTime * 500;   %converts time to row number
LastRowNumber = EndTime * 500;      %converts time to row number

Submatrix = matrix(FirstRowNumber:LastRowNumber,:); %defines submatrix

[m,n] = size(Submatrix);

if (angle_input == 2)

   k=1;
   j=1;

   while (j <= m)

      rps = Submatrix(k,2);
      theta = Submatrix(k,29);

      if  (theta < theta1) | (theta > theta2)

         Submatrix(k,:) = [];
```

```
        else

            k = k+1;

        end

        j = j+1;

    end

end

%DEFINE THE SIZE OF THE NEW SUBMATRIX----------------------------------

[m,n] = size(Submatrix);

%DEFINE WHICH VARIABLES TO PLOT ON THE X-AXIS--------------------------

Xaxis = input('Select variable to plot on x-axis: ');

if (Xaxis == 1)

    x(1:m,1) = Submatrix(1:m,1);

end

%DEFINE WHICH VARIABLES TO PLOT ON THE Y-AXIS--------------------------

%USER DEFINES EITHER GLOBAL, AFT, FORWARD FORCES-----------------------

Yaxis = input('Select forces to plot (global=1, aft=2, forward=3): ');

y(1:m,1) = Submatrix(1:m,29);

if (Yaxis == 1)

    Fx1(1:m,1) = Submatrix(1:m,4);
    Fx2(1:m,1) = Submatrix(1:m,5);
    Fy1(1:m,1) = Submatrix(1:m,6);
    Fz1(1:m,1) = Submatrix(1:m,7);
    Fz2(1:m,1) = Submatrix(1:m,8);
    Fz3(1:m,1) = Submatrix(1:m,9);

    %DETERMINE THE MEAN AND STANDARD DEVIATIONS FOR EACH SET OF FORCES----

    Fx1_ave = mean(Fx1);
    Fx2_ave = mean(Fx2);
    Fy1_ave = mean(Fy1);
    Fz1_ave = mean(Fz1);
    Fz2_ave = mean(Fz2);
    Fz3_ave = mean(Fz3);

    Fx1_std = std(Fx1);
    Fx2_std = std(Fx2);
    Fy1_std = std(Fy1);
    Fz1_std = std(Fz1);
    Fz2_std = std(Fz2);
    Fz3_std = std(Fz3);
```

```
disp('            Mean Value   Standard Deviation')
fprintf(1,' \t Fx1 \t\t %6.1f \t\t %6.1f \n',Fx1_ave,Fx1_std);
fprintf(1,' \t Fx2 \t\t %6.1f \t\t %6.1f \n',Fx2_ave,Fx2_std);
fprintf(1,' \t Fx1 \t\t %6.1f \t\t %6.1f \n',Fy1_ave,Fy1_std);
fprintf(1,' \t Fz1 \t\t %6.1f \t\t %6.1f \n',Fz1_ave,Fz1_std);
fprintf(1,' \t Fz2 \t\t %6.1f \t\t %6.1f \n',Fz2_ave,Fz2_std);
fprintf(1,' \t Fz3 \t\t %6.1f \t\t %6.1f \n',Fz3_ave,Fz3_std);

%DETERMINE THE MINIMUM AND MAXIMUM FORCES ON THE DYNO------------------

Fx1max = max(Fx1);
Fx2max = max(Fx2);
Fy1max = max(Fy1);
Fz1max = max(Fz1);
Fz2max = max(Fz2);
Fz3max = max(Fz3);

Fx1min = min(Fx1);
Fx2min = min(Fx2);
Fy1min = min(Fy1);
Fz1min = min(Fz1);
Fz2min = min(Fz2);
Fz3min = min(Fz3);

[Fx1max_r,Fx1max_c] = find(Submatrix(:,4)==Fx1max);
[Fx2max_r,Fx2max_c] = find(Submatrix(:,5)==Fx2max);
[Fy1max_r,Fy1max_c] = find(Submatrix(:,6)==Fy1max);
[Fz1max_r,Fz1max_c] = find(Submatrix(:,7)==Fz1max);
[Fz2max_r,Fz2max_c] = find(Submatrix(:,8)==Fz2max);
[Fz3max_r,Fz3max_c] = find(Submatrix(:,9)==Fz3max);

theta_Fx1max = Submatrix(Fx1max_r,29);
theta_Fx2max = Submatrix(Fx2max_r,29);
theta_Fy1max = Submatrix(Fy1max_r,29);
theta_Fz1max = Submatrix(Fz1max_r,29);
theta_Fz2max = Submatrix(Fz2max_r,29);
theta_Fz3max = Submatrix(Fz3max_r,29);

[Fx1min_r,Fx1min_c] = find(Submatrix(:,4)==Fx1min);
[Fx2min_r,Fx2min_c] = find(Submatrix(:,5)==Fx2min);
[Fy1min_r,Fy1min_c] = find(Submatrix(:,6)==Fy1min);
[Fz1min_r,Fz1min_c] = find(Submatrix(:,7)==Fz1min);
[Fz2min_r,Fz2min_c] = find(Submatrix(:,8)==Fz2min);
[Fz3min_r,Fz3min_c] = find(Submatrix(:,9)==Fz3min);

theta_Fx1min = Submatrix(Fx1min_r,29);
theta_Fx2min = Submatrix(Fx2min_r,29);
theta_Fy1min = Submatrix(Fy1min_r,29);
theta_Fz1min = Submatrix(Fz1min_r,29);
theta_Fz2min = Submatrix(Fz2min_r,29);
theta_Fz3min = Submatrix(Fz3min_r,29);

disp('            Maximum Value @ Angle       Minimum Value @ Angle')
fprintf(1,' \t Fx1 \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Fx1max,theta_Fx1max,Fx1min,theta_Fx1min)
fprintf(1,' \t Fx2 \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Fx2max,theta_Fx2max,Fx2min,theta_Fx2min)
```

```
   fprintf(1,' \t Fy1 \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Fy1max,theta_Fy1max,Fy1min,theta_Fy1min)
   fprintf(1,' \t Fz1 \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Fz1max,theta_Fz1max,Fz1min,theta_Fz1min)
   fprintf(1,' \t Fz2 \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Fz2max,theta_Fz2max,Fz2min,theta_Fz2min)
   fprintf(1,' \t Fz3 \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Fz3max,theta_Fz3max,Fz3min,theta_Fz3min)

   %DETERMINE THE TOTAL FORCES AND MOMENTS ON THE DYNO------------------------

   Fx_tot = Fx1 + Fx2;
   Fy_tot = Fy1;
   Fz_tot = Fz1 + Fz2 + Fz3;

   Fx_tot_max = max(Fx_tot);
   Fy_tot_max = max(Fy_tot);
   Fz_tot_max = max(Fz_tot);

   F = [Fx1, Fx2, Fy1, Fz1, Fz2, Fz3];

   Moment_matrix = Moments(F);

   Mx = Moment_matrix(:,1);
   My = Moment_matrix(:,2);
   Mz = Moment_matrix(:,3);

   Mx_max = max(Mx);
   My_max = max(My);
   Mz_max = max(Mz);

   [Fx_tot_max_r,Fx_tot_max_c] = find(Fx_tot==Fx_tot_max);
   [Fy_tot_max_r,Fy_tot_max_c] = find(Fy_tot==Fy_tot_max);
   [Fz_tot_max_r,Fz_tot_max_c] = find(Fz_tot==Fz_tot_max);
   [Mx_max_r,Mx_Max_c] = find(Mx==Mx_max);
   [My_max_r,My_Max_c] = find(My==My_max);
   [Mz_max_r,Mz_Max_c] = find(Mz==Mz_max);

   theta_Fxtot_max = Submatrix(Fx_tot_max_r,29);
   theta_Fytot_max = Submatrix(Fy_tot_max_r,29);
   theta_Fztot_max = Submatrix(Fz_tot_max_r,29);
   theta_Mxtot_max = Submatrix(Mx_max_r,29);
   theta_Mytot_max = Submatrix(My_max_r,29);
   theta_Mztot_max = Submatrix(Mz_max_r,29);

   Fx_tot_min = min(Fx_tot);
   Fy_tot_min = min(Fy_tot);
   Fz_tot_min = min(Fz_tot);
   Mx_min = min(Mx);
   My_min = min(My);
   Mz_min = min(Mz);

   [Fx_tot_min_r,Fx_tot_min_c] = find(Fx_tot==Fx_tot_min);
   [Fy_tot_min_r,Fy_tot_min_c] = find(Fy_tot==Fy_tot_min);
   [Fz_tot_min_r,Fz_tot_min_c] = find(Fz_tot==Fz_tot_min);
   [Mx_min_r,Mx_Min_c] = find(Mx==Mx_min);
   [My_min_r,My_Min_c] = find(My==My_min);
   [Mz_min_r,Mz_Min_c] = find(Mz==Mz_min);
```

```
theta_Fxtot_min = Submatrix(Fx_tot_min_r,29);
theta_Fytot_min = Submatrix(Fy_tot_min_r,29);
theta_Fztot_min = Submatrix(Fz_tot_min_r,29);
theta_Mxtot_min = Submatrix(Mx_min_r,29);
theta_Mytot_min = Submatrix(My_min_r,29);
theta_Mztot_min = Submatrix(Mz_min_r,29);

disp('          Maximum Total Value @ Angle    Minimum Total Value @ Angle')
fprintf(1,' \t Fx \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Fx_tot_max,theta_Fxtot_max,Fx_tot_min,theta_Fxtot_min);
fprintf(1,' \t Fy \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Fy_tot_max,theta_Fytot_max,Fy_tot_min,theta_Fytot_min);
fprintf(1,' \t Fz \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Fz_tot_max,theta_Fztot_max,Fz_tot_min,theta_Fztot_min);
fprintf(1,' \t Mx \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Mx_max,theta_Mxtot_max,Mx_min,theta_Mxtot_min);
fprintf(1,' \t My \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',My_max,theta_Mytot_max,My_min,theta_Mytot_min);
fprintf(1,' \t Mz \t\t %6.1f @ %6.1f deg \t\t %6.1f @ %6.1f deg
\n',Mz_max,theta_Mztot_max,Mz_min,theta_Mztot_min);

%DETERMINE THE MEAN AND STANDARD DEVIATIONS FOR EACH SET OF TOTAL FORCES

Fx_tot_ave = mean(Fx_tot);
Fy_tot_ave = mean(Fy_tot);
Fz_tot_ave = mean(Fz_tot);
Mx_ave = mean(Mx);
My_ave = mean(My);
Mz_ave = mean(Mz);

Fx_tot_std = std(Fx_tot);
Fy_tot_std = std(Fy_tot);
Fz_tot_std = std(Fz_tot);
Mx_std = std(Mx);
My_std = std(My);
Mz_std = std(Mz);

disp('          Mean Value   Standard Deviation (TOTAL VALUES)')
fprintf(1,' \t Fx \t\t %6.1f \t\t %6.1f \n',Fx_tot_ave,Fx_tot_std);
fprintf(1,' \t Fy \t\t %6.1f \t\t %6.1f \n',Fy_tot_ave,Fy_tot_std);
fprintf(1,' \t Fz \t\t %6.1f \t\t %6.1f \n',Fz_tot_ave,Fz_tot_std);
fprintf(1,' \t Mx \t\t %6.1f \t\t %6.1f \n',Mx_ave,Mx_std);
fprintf(1,' \t My \t\t %6.1f \t\t %6.1f \n',My_ave,My_std);
fprintf(1,' \t Mz \t\t %6.1f \t\t %6.1f \n',Mz_ave,Mz_std);

%DETERMINE THE MAXIMUM DIFFERENCE BETWEEN MATRIX ELEMENTS-------------------

max_diff = find_maxdiff(Submatrix,Fx1,Fx2,Fy1,Fz1,Fz2,Fz3);

disp('          Maximum Difference @ Angle')
fprintf(1,' \t Fx1 \t\t %6.1f @ %6.1f deg \n',max_diff(1,1),max_diff(2,1));
fprintf(1,' \t Fx2 \t\t %6.1f @ %6.1f deg \n',max_diff(1,2),max_diff(2,2));
fprintf(1,' \t Fy  \t\t %6.1f @ %6.1f deg \n',max_diff(1,3),max_diff(2,3));
fprintf(1,' \t Fz1 \t\t %6.1f @ %6.1f deg \n',max_diff(1,4),max_diff(2,4));
fprintf(1,' \t Fz2 \t\t %6.1f @ %6.1f deg \n',max_diff(1,5),max_diff(2,5));
fprintf(1,' \t Fz3 \t\t %6.1f @ %6.1f deg \n',max_diff(1,6),max_diff(2,6));
```

```
figure;

subplot(3,1,1); [AX,H1,H2] = plotyy(x,Fx1,x,y,'plot');
title('Global Dyno Forces, Fx');
xlabel('Time (s)');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');
set(AX(2),'YLim',[0 360]);
set(AX(2),'ytick',[0,90,180,270,360]);
set(AX(2),'YGrid','on');

subplot(3,1,2); [AX,H1,H2] = plotyy(x,Fx2,x,y,'plot');
title('Global Dyno Forces, Fx2');
xlabel('Time (s)');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');
set(AX(2),'YLim',[0 360]);
set(AX(2),'ytick',[0,90,180,270,360]);
set(AX(2),'YGrid','on');

subplot(3,1,3); [AX,H1,H2] = plotyy(x,Fy1,x,y,'plot');
title('Global Dyno Forces, Fy');
xlabel('Time (s)');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');
xlabel('Time (s)');
set(AX(2),'YLim',[0 360]);
set(AX(2),'ytick',[0,90,180,270,360]);
set(AX(2),'YGrid','on');

figure;

subplot(3,1,1); [AX,H1,H2] = plotyy(x,Fz1,x,y,'plot');
title('Global Dyno Forces, Fz1');
xlabel('Time (s)');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');
set(AX(2),'YLim',[0 360]);
set(AX(2),'ytick',[0,90,180,270,360]);
set(AX(2),'YGrid','on');

subplot(3,1,2); [AX,H1,H2] = plotyy(x,Fz2,x,y,'plot');
```

```
    title('Global Dyno Forces, Fz2');
    xlabel('Time (s)');
    axes(AX(1));
    ylabel('Force (N)');
    axes(AX(2));
    ylabel('Theta (deg)');
    set(H1,'LineStyle','--');
    set(H2,'LineStyle',':');
    set(AX(2),'YLim',[0 360]);
    set(AX(2),'ytick',[0,90,180,270,360]);
    set(AX(2),'YGrid','on');

    subplot(3,1,3); [AX,H1,H2] = plotyy(x,Fz3,x,y,'plot');
    title('Global Dyno Forces, Fz3');
    xlabel('Time (s)');
    axes(AX(1));
    ylabel('Force (N)');
    axes(AX(2));
    ylabel('Theta (deg)');
    set(H1,'LineStyle','--');
    set(H2,'LineStyle',':');
    set(AX(2),'YLim',[0 360]);
    set(AX(2),'ytick',[0,90,180,270,360]);
    set(AX(2),'YGrid','on');

elseif (Yaxis == 2)

    y1(1:m,1) = Submatrix(1:m,10);
    y2(1:m,1) = Submatrix(1:m,11);
    y3(1:m,1) = Submatrix(1:m,12);
    y4(1:m,1) = Submatrix(1:m,13);
    y5(1:m,1) = Submatrix(1:m,14);
    y6(1:m,1) = Submatrix(1:m,15);

    %DETERMINE THE MEAN AND STANDARD DEVIATIONS FOR EACH SET OF FORCES----

    y1_ave = mean(y1);
    y2_ave = mean(y2);
    y3_ave = mean(y3);
    y4_ave = mean(y4);
    y5_ave = mean(y5);
    y6_ave = mean(y6);

    y1_std = std(y1);
    y2_std = std(y2);
    y3_std = std(y3);
    y4_std = std(y4);
    y5_std = std(y5);
    y6_std = std(y6);

    disp('            Mean Value   Standard Deviation')
    fprintf(1,' \t Fx  \t\t %6.1f \t\t %6.1f \n',y1_ave,y1_std);
    fprintf(1,' \t Fy  \t\t %6.1f \t\t %6.1f \n',y2_ave,y2_std);
    fprintf(1,' \t Fz  \t\t %6.1f \t\t %6.1f \n',y3_ave,y3_std);
    fprintf(1,' \t Mx  \t\t %6.1f \t\t %6.1f \n',y4_ave,y4_std);
    fprintf(1,' \t My  \t\t %6.1f \t\t %6.1f \n',y5_ave,y5_std);
    fprintf(1,' \t Mz  \t\t %6.1f \t\t %6.1f \n',y6_ave,y6_std);
```

```
%DETERMINE THE MINIMUM AND MAXIMUM FORCES ON THE DYNO------------------

y1max = max(y1);
y2max = max(y2);
y3max = max(y3);
y4max = max(y4);
y5max = max(y5);
y6max = max(y6);

y1min = min(y1);
y2min = min(y2);
y3min = min(y3);
y4min = min(y4);
y5min = min(y5);
y6min = min(y6);

[y1max_r,y1max_c] = find(Submatrix(:,10)==y1max);
[y2max_r,y2max_c] = find(Submatrix(:,11)==y2max);
[y3max_r,y3max_c] = find(Submatrix(:,12)==y3max);
[y4max_r,y4max_c] = find(Submatrix(:,13)==y4max);
[y5max_r,y5max_c] = find(Submatrix(:,14)==y5max);
[y6max_r,y6max_c] = find(Submatrix(:,15)==y6max);

theta_y1max = Submatrix(y1max_r,29);
theta_y2max = Submatrix(y2max_r,29);
theta_y3max = Submatrix(y3max_r,29);
theta_y4max = Submatrix(y4max_r,29);
theta_y5max = Submatrix(y5max_r,29);
theta_y6max = Submatrix(y6max_r,29);

[y1min_r,y1min_c] = find(Submatrix(:,10)==y1min);
[y2min_r,y2min_c] = find(Submatrix(:,11)==y2min);
[y3min_r,y3min_c] = find(Submatrix(:,12)==y3min);
[y4min_r,y4min_c] = find(Submatrix(:,13)==y4min);
[y5min_r,y5min_c] = find(Submatrix(:,14)==y5min);
[y6min_r,y6min_c] = find(Submatrix(:,15)==y6min);

theta_y1min = Submatrix(y1min_r,29);
theta_y2min = Submatrix(y2min_r,29);
theta_y3min = Submatrix(y3min_r,29);
theta_y4min = Submatrix(y4min_r,29);
theta_y5min = Submatrix(y5min_r,29);
theta_y6min = Submatrix(y6min_r,29);

disp('            Maximum Value @ Angle')
fprintf(1,' \t Fx \t\t %6.1f @ %6.1f deg \n',y1max,theta_y1max)
fprintf(1,' \t Fy \t\t %6.1f @ %6.1f deg \n',y2max,theta_y2max)
fprintf(1,' \t Fz \t\t %6.1f @ %6.1f deg \n',y3max,theta_y3max)
fprintf(1,' \t Mx \t\t %6.1f @ %6.1f deg \n',y4max,theta_y4max)
fprintf(1,' \t My \t\t %6.1f @ %6.1f deg \n',y5max,theta_y5max)
fprintf(1,' \t Mz \t\t %6.1f @ %6.1f deg \n',y6max,theta_y6max)

disp('            Minimum Value @ Angle')
fprintf(1,' \t Fx \t\t %6.1f @ %6.1f deg \n',y1min,theta_y1min)
fprintf(1,' \t Fy \t\t %6.1f @ %6.1f deg \n',y2min,theta_y2min)
fprintf(1,' \t Fz \t\t %6.1f @ %6.1f deg \n',y3min,theta_y3min)
fprintf(1,' \t Mx \t\t %6.1f @ %6.1f deg \n',y4min,theta_y4min)
fprintf(1,' \t My \t\t %6.1f @ %6.1f deg \n',y5min,theta_y5min)
```

```
fprintf(1,' \t Mz \t\t %6.1f @ %6.1f deg \n',y6min,theta_y6min)

%DETERMINE THE ABSOLUTE MAXIMUM FORCES ON THE DYNO------------------

y1max_abs = abs(y1max);
y2max_abs = abs(y2max);
y3max_abs = abs(y3max);
y4max_abs = abs(y4max);
y5max_abs = abs(y5max);
y6max_abs = abs(y6max);

y1min_abs = abs(y1min);
y2min_abs = abs(y2min);
y3min_abs = abs(y3min);
y4min_abs = abs(y4min);
y5min_abs = abs(y5min);
y6min_abs = abs(y6min);

if(y1min_abs > y1max_abs)

   y1max_abs = y1min;
   theta_y1max = theta_y1min;

end

if(y2min_abs > y2max_abs)

   y2max_abs = y2min;
   theta_y2max = theta_y2min;

end

if(y3min_abs > y3max_abs)

   y3max_abs = y3min;
   theta_y3max = theta_y3min;

end

if(y4min_abs > y4max_abs)

   y4max_abs = y4min;
   theta_y4max = theta_y4min;

end

if(y5min_abs > y5max_abs)

   y5max_abs = y5min;
   theta_y5max = theta_y5min;

end

if(y6min_abs > y6max_abs)

   y6max_abs = y6min;
   theta_y6max = theta_y6min;
```

```
end

disp('          Absolute Maximum Value @ Angle');
fprintf(1,' \t Fx \t\t %6.1f @ %6.1f deg \n',y1max_abs,theta_y1max);
fprintf(1,' \t Fy \t\t %6.1f @ %6.1f deg \n',y2max_abs,theta_y2max);
fprintf(1,' \t Fz \t\t %6.1f @ %6.1f deg \n',y3max_abs,theta_y3max);
fprintf(1,' \t Mx \t\t %6.1f @ %6.1f deg \n',y4max_abs,theta_y4max);
fprintf(1,' \t My \t\t %6.1f @ %6.1f deg \n',y5max_abs,theta_y5max);
fprintf(1,' \t Mz \t\t %6.1f @ %6.1f deg \n',y6max_abs,theta_y6max);

%DETERMINE THE MAXIMUM DIFFERENCE BETWEEN MATRIX ELEMENTS-------------------

max_diff = find_maxdiff(Submatrix,y1,y2,y3,y4,y5,y6);

disp('            Maximum Difference @ Angle')
fprintf(1,' \t Fx  \t\t %6.1f @ %6.1f deg \n',max_diff(1,1),max_diff(2,1));
fprintf(1,' \t Fy  \t\t %6.1f @ %6.1f deg \n',max_diff(1,2),max_diff(2,2));
fprintf(1,' \t Fz  \t\t %6.1f @ %6.1f deg \n',max_diff(1,3),max_diff(2,3));
fprintf(1,' \t Mx  \t\t %6.1f @ %6.1f deg \n',max_diff(1,4),max_diff(2,4));
fprintf(1,' \t My  \t\t %6.1f @ %6.1f deg \n',max_diff(1,5),max_diff(2,5));
fprintf(1,' \t Mz  \t\t %6.1f @ %6.1f deg \n',max_diff(1,6),max_diff(2,6));

figure;

subplot(3,1,1); [AX,H1,H2] = plotyy(x,y1,x,y,'plot');
title('Aft Dyno Forces, Fx');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');


subplot(3,1,2); [AX,H1,H2] = plotyy(x,y2,x,y,'plot');
title('Aft Dyno Forces, Fy');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');


subplot(3,1,3); [AX,H1,H2] = plotyy(x,y3,x,y,'plot');
title('Aft Dyno Forces, Fz');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');
xlabel('Time (s)');

figure;

subplot(3,1,1); [AX,H1,H2] = plotyy(x,y4,x,y,'plot');
title('Aft Dyno Moments, Mx');
```

```
  axes(AX(1));
  ylabel('Moments (Nm)');
  axes(AX(2));
  ylabel('Theta (deg)');
  set(H1,'LineStyle','--');
  set(H2,'LineStyle',':');

  subplot(3,1,2); [AX,H1,H2] = plotyy(x,y5,x,y,'plot');
  title('Aft Dyno Moments, My');
  axes(AX(1));
  ylabel('Moments (Nm)');
  axes(AX(2));
  ylabel('Theta (deg)');
  set(H1,'LineStyle','--');
  set(H2,'LineStyle',':');

  subplot(3,1,3); [AX,H1,H2] = plotyy(x,y6,x,y,'plot');
  title('Aft Dyno Moments, Mz');
  axes(AX(1));
  ylabel('Moments (Nm)');
  axes(AX(2));
  ylabel('Theta (deg)');
  set(H1,'LineStyle','--');
  set(H2,'LineStyle',':');
  xlabel('Time (s)');

elseif (Yaxis == 3)

  y1(1:m,1) = Submatrix(1:m,16);
  y2(1:m,1) = Submatrix(1:m,17);
  y3(1:m,1) = Submatrix(1:m,18);
  y4(1:m,1) = Submatrix(1:m,19);
  y5(1:m,1) = Submatrix(1:m,20);
  y6(1:m,1) = Submatrix(1:m,21);

  %DETERMINE THE MEAN AND STANDARD DEVIATIONS FOR EACH SET OF FORCES----

  y1_ave = mean(y1);
  y2_ave = mean(y2);
  y3_ave = mean(y3);
  y4_ave = mean(y4);
  y5_ave = mean(y5);
  y6_ave = mean(y6);

  y1_std = std(y1);
  y2_std = std(y2);
  y3_std = std(y3);
  y4_std = std(y4);
  y5_std = std(y5);
  y6_std = std(y6);

  disp('            Mean Value   Standard Deviation')
  fprintf(1,' \t Fx  \t\t %6.1f \t\t %6.1f \n',y1_ave,y1_std);
  fprintf(1,' \t Fy  \t\t %6.1f \t\t %6.1f \n',y2_ave,y2_std);
  fprintf(1,' \t Fz  \t\t %6.1f \t\t %6.1f \n',y3_ave,y3_std);
  fprintf(1,' \t Mx  \t\t %6.1f \t\t %6.1f \n',y4_ave,y4_std);
  fprintf(1,' \t My  \t\t %6.1f \t\t %6.1f \n',y5_ave,y5_std);
  fprintf(1,' \t Mz  \t\t %6.1f \t\t %6.1f \n',y6_ave,y6_std);
```

```
%DETERMINE THE MINIMUM AND MAXIMUM FORCES ON THE DYNO------------------

y1max = max(y1);
y2max = max(y2);
y3max = max(y3);
y4max = max(y4);
y5max = max(y5);
y6max = max(y6);

y1min = min(y1);
y2min = min(y2);
y3min = min(y3);
y4min = min(y4);
y5min = min(y5);
y6min = min(y6);

[y1max_r,y1max_c] = find(Submatrix(:,16)==y1max);
[y2max_r,y2max_c] = find(Submatrix(:,17)==y2max);
[y3max_r,y3max_c] = find(Submatrix(:,18)==y3max);
[y4max_r,y4max_c] = find(Submatrix(:,19)==y4max);
[y5max_r,y5max_c] = find(Submatrix(:,20)==y5max);
[y6max_r,y6max_c] = find(Submatrix(:,21)==y6max);

theta_y1max = Submatrix(y1max_r,29);
theta_y2max = Submatrix(y2max_r,29);
theta_y3max = Submatrix(y3max_r,29);
theta_y4max = Submatrix(y4max_r,29);
theta_y5max = Submatrix(y5max_r,29);
theta_y6max = Submatrix(y6max_r,29);

[y1min_r,y1min_c] = find(Submatrix(:,16)==y1min);
[y2min_r,y2min_c] = find(Submatrix(:,17)==y2min);
[y3min_r,y3min_c] = find(Submatrix(:,18)==y3min);
[y4min_r,y4min_c] = find(Submatrix(:,19)==y4min);
[y5min_r,y5min_c] = find(Submatrix(:,20)==y5min);
[y6min_r,y6min_c] = find(Submatrix(:,21)==y6min);

theta_y1min = Submatrix(y1min_r,29);
theta_y2min = Submatrix(y2min_r,29);
theta_y3min = Submatrix(y3min_r,29);
theta_y4min = Submatrix(y4min_r,29);
theta_y5min = Submatrix(y5min_r,29);
theta_y6min = Submatrix(y6min_r,29);

disp('              Maximum Value @ Angle')
fprintf(1,' \t Fx \t\t %6.1f @ %6.1f deg \n',y1max,theta_y1max)
fprintf(1,' \t Fy \t\t %6.1f @ %6.1f deg \n',y2max,theta_y2max)
fprintf(1,' \t Fz \t\t %6.1f @ %6.1f deg \n',y3max,theta_y3max)
fprintf(1,' \t Mx \t\t %6.1f @ %6.1f deg \n',y4max,theta_y4max)
fprintf(1,' \t My \t\t %6.1f @ %6.1f deg \n',y5max,theta_y5max)
fprintf(1,' \t Mz \t\t %6.1f @ %6.1f deg \n',y6max,theta_y6max)

disp('              Minimum Value @ Angle')
fprintf(1,' \t Fx \t\t %6.1f @ %6.1f deg \n',y1min,theta_y1min)
fprintf(1,' \t Fy \t\t %6.1f @ %6.1f deg \n',y2min,theta_y2min)
fprintf(1,' \t Fz \t\t %6.1f @ %6.1f deg \n',y3min,theta_y3min)
fprintf(1,' \t Mx \t\t %6.1f @ %6.1f deg \n',y4min,theta_y4min)
```

```
fprintf(1,' \t My \t\t %6.1f @ %6.1f deg \n',y5min,theta_y5min)
fprintf(1,' \t Mz \t\t %6.1f @ %6.1f deg \n',y6min,theta_y6min)

   %DETERMINE THE ABSOLUTE MAXIMUM FORCES ON THE DYNO------------------

y1max_abs = abs(y1max);
y2max_abs = abs(y2max);
y3max_abs = abs(y3max);
y4max_abs = abs(y4max);
y5max_abs = abs(y5max);
y6max_abs = abs(y6max);

y1min_abs = abs(y1min);
y2min_abs = abs(y2min);
y3min_abs = abs(y3min);
y4min_abs = abs(y4min);
y5min_abs = abs(y5min);
y6min_abs = abs(y6min);

if(y1min_abs > y1max_abs)

   y1max_abs = y1min;
   theta_y1max = theta_y1min;

end

if(y2min_abs > y2max_abs)

   y2max_abs = y2min;
   theta_y2max = theta_y2min;

end

if(y3min_abs > y3max_abs)

   y3max_abs = y3min;
   theta_y3max = theta_y3min;

end

if(y4min_abs > y4max_abs)

   y4max_abs = y4min;
   theta_y4max = theta_y4min;

end

if(y5min_abs > y5max_abs)

   y5max_abs = y5min;
   theta_y5max = theta_y5min;

end

if(y6min_abs > y6max_abs)

   y6max_abs = y6min;
   theta_y6max = theta_y6min;
```

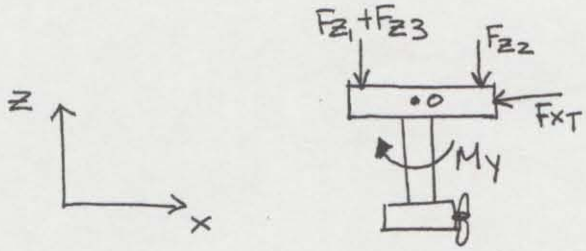
```
end

disp('          Absolute Maximum Value @ Angle');
fprintf(1,' \t Fx \t\t %6.1f @ %6.1f deg \n',y1max_abs,theta_y1max);
fprintf(1,' \t Fy \t\t %6.1f @ %6.1f deg \n',y2max_abs,theta_y2max);
fprintf(1,' \t Fz \t\t %6.1f @ %6.1f deg \n',y3max_abs,theta_y3max);
fprintf(1,' \t Mx \t\t %6.1f @ %6.1f deg \n',y4max_abs,theta_y4max);
fprintf(1,' \t My \t\t %6.1f @ %6.1f deg \n',y5max_abs,theta_y5max);
fprintf(1,' \t Mz \t\t %6.1f @ %6.1f deg \n',y6max_abs,theta_y6max);

%DETERMINE THE MAXIMUM DIFFERENCE BETWEEN MATRIX ELEMENTS-------------------

max_diff = find_maxdiff(Submatrix,y1,y2,y3,y4,y5,y6);

disp('             Maximum Difference @ Angle')
fprintf(1,' \t Fx  \t\t %6.1f @ %6.1f deg \n',max_diff(1,1),max_diff(2,1));
fprintf(1,' \t Fy  \t\t %6.1f @ %6.1f deg \n',max_diff(1,2),max_diff(2,2));
fprintf(1,' \t Fz  \t\t %6.1f @ %6.1f deg \n',max_diff(1,3),max_diff(2,3));
fprintf(1,' \t Mx  \t\t %6.1f @ %6.1f deg \n',max_diff(1,4),max_diff(2,4));
fprintf(1,' \t My  \t\t %6.1f @ %6.1f deg \n',max_diff(1,5),max_diff(2,5));
fprintf(1,' \t Mz  \t\t %6.1f @ %6.1f deg \n',max_diff(1,6),max_diff(2,6));

figure;

subplot(3,1,1); [AX,H1,H2] = plotyy(x,y1,x,y,'plot');
title('Forward Dyno Forces, Fx');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');


subplot(3,1,2); [AX,H1,H2] = plotyy(x,y2,x,y,'plot');
title('Forward Dyno Forces, Fy');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');

subplot(3,1,3); [AX,H1,H2] = plotyy(x,y3,x,y,'plot');
title('Forward Dyno Forces, Fz');
axes(AX(1));
ylabel('Force (N)');
axes(AX(2));
ylabel('Theta (deg)');
set(H1,'LineStyle','--');
set(H2,'LineStyle',':');
xlabel('Time (s)');

figure;

subplot(3,1,1); [AX,H1,H2] = plotyy(x,y4,x,y,'plot');
title('Forward Dyno Moments, Mx');
axes(AX(1));
```

```
  ylabel('Moments (Nm)');
  axes(AX(2));
  ylabel('Theta (deg)');
  set(H1,'LineStyle','--');
  set(H2,'LineStyle',':');

  subplot(3,1,2); [AX,H1,H2] = plotyy(x,y5,x,y,'plot');
  title('Forward Dyno Moments, My');
  axes(AX(1));
  ylabel('Moments (Nm)');
  axes(AX(2));
  ylabel('Theta (deg)');
  set(H1,'LineStyle','--');
  set(H2,'LineStyle',':');

  subplot(3,1,3); [AX,H1,H2] = plotyy(x,y6,x,y,'plot');
  title('Forward Dyno Moments, Mz');
  axes(AX(1));
  ylabel('Moments (Nm)');
  axes(AX(2));
  ylabel('Theta (deg)');
  set(H1,'LineStyle','--');
  set(H2,'LineStyle',':');
  xlabel('Time (s)');

end
```

Appendix B

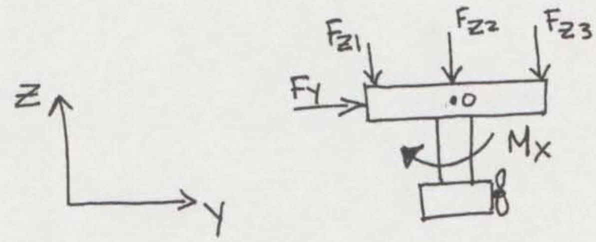
```
%returns the vector [Mx, My, Mz] when the vector
%F = [Fx1,Fx2,Fy,Fz1,Fz2,Fz3] is entered
function M=Moments(F)

[m,n] = size(F);

Xz1=0.1524;
Yz1=0.2640;
Zz1=0.1968;
Xz2=0.3048;
Yz2=0;
Zz2=Zz1;
Xz3=Xz1;
Yz3=Yz1;
Zz3=Zz1;
Xx1=0.3134;
Yx1=.2640;
Zx1=0.0863;
Xy=0;
Yy=0.2640;
Zy=0.0863;

for i = 1:m

Mo(i,1)=Yz1*(F(i,4)-F(i,6));
Mo(i,2)=(F(i,4)+F(i,6))*Xz1-F(i,5);
Mo(i,3)=Yx1*(F(i,2)-F(i,1));

end

M = Mo;
%returns the vector [Mx, My, Mz]
```

Equations for Moments.m



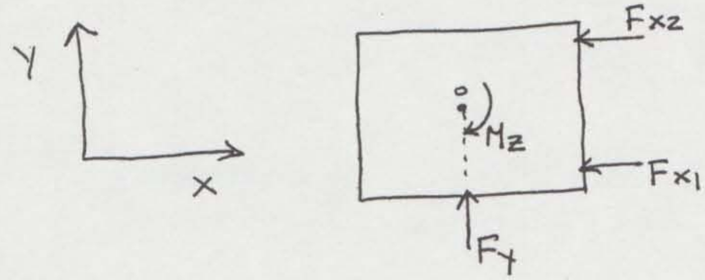
$$+\circlearrowleft\ \Sigma M_o = 0 = M_y + F_{z2}X_{z2} - (F_{z1}+F_{z3})X_{z1}$$

$$M_y = (F_{z1}+F_{z3})X_{z1} - F_{z2}X_{z2}$$



$$+\circlearrowright\ \Sigma M_o = 0 = M_x + F_{z3}Y_{z3} - F_{z1}Y_{z1}$$

$$M_x = Y_{z1}(F_{z1} - F_{z3})$$



$$+\circlearrowright\ M_o = 0 = M_z + F_{x1}Y_{x1} - F_{x2}Y_{x2}$$

$$M_z = Y_{x1}(F_{x2} - F_{x1})$$

Appendix C

```
%Program to determine the calibration coefficients
clear all
clc

%Enter total number of files (number of trials in the setup)
totalfiles = 6;
%Enter the starting number for the files
starti = 1;
%Enter the name of the source directory
source_directory = 'C:\TEMP\Test Data\' ;
%Enter the common part in the source file names. For example:
%for files img00001.bmp to img00512.bmp        enter img00
source_file_common_partP= 'P';
source_file_common_partR= 'R';
source_file_common_part_coord= 'coord';
%Enter the extension of the files
fname3 = '.dat';

%==================================================================
% End of input

j=1;

for i = starti : (starti+totalfiles-1)

   %File names
   fname2 = num2str(i);

   if (i < 10 )
      fname1 = '00';
      fnameP  = strcat(source_directory, source_file_common_partP, fname1, fname2, fname3);
      fnameR  = strcat(source_directory, source_file_common_partR, fname1, fname2, fname3);
      fnameCoord  = strcat(source_directory, source_file_common_part_coord, fname1,fname2, fname3);
   elseif (i <= 99)
      fname1 = '0';
      fnameP  = strcat(source_directory, source_file_common_partP, fname1, fname2, fname3);
      fnameR  = strcat(source_directory, source_file_common_partR, fname1, fname2, fname3);
      fnameCoord  = strcat(source_directory, source_file_common_part_coord, fname1, fname3);
   else
      fnameP  = strcat(source_directory, source_file_common_partP, fname2, fname3);
      fnameR  = strcat(source_directory, source_file_common_partR, fname2, fname3);
      fnameCoord  = strcat(source_directory, source_file_common_part_coord, fname3);
   end

   %open files containing P=Px,Py,Pz -> load applied on the system
   P=importdata (fnameP) ;

   %c=(x1,y1,z1; x2,y2,z2; x3,y3,z3) -> location of the load applied
   % x1,y1,z1 is the point where Px was applied and so on
   c=importdata (fnameCoord);


   %open file containing R = Rx1,Rx2,Ry,Rz1,Rz2,Rz3 -> output measured by the individual dynos
   R=importdata (fnameR);

   %Calculate the forces and calibration matrix
```

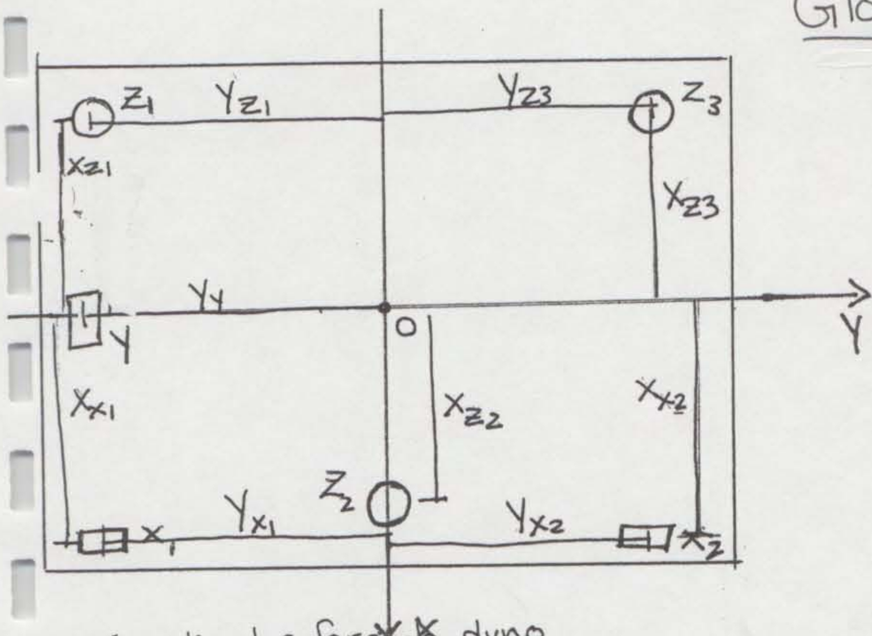
```
    MtxF(:,j)=ForceFinder(P,c);

    MtxR(:,j)=R';

    %Moments in x, y and z
    MtxM(:,j)=Moments(MtxF(:,j));

    j=j+1;
end

MtxCalib=MtxR*inv(MtxF)
```

Appendix D

```
%Function to determine the forces on each dyno resulting from a global force
function F=ForceFinder(P,C)

%R is a vector with the components of the load P
%C is the point of application of the load

%Constants
Xz1=0.1524;
Yz1=0.2640;
Zz1=0.1968;
Xz2=0.3048;
Yz2=0;
Zz2=Zz1;
Xz3=Xz1;
Yz3=Yz1;
Zz3=Zz1;
Xx1=0.3134;
Yx1=.2640;
Zx1=0.0863;
Xy=0;
Yy=0.2640;
Zy=0.0863;

%System of equations
R=[P(1) 0   0;
   0  P(2) 0;
   0   0  P(3)];

for i=1:3

   A=[ 1    1  0  0  0    0;
      0   0  1  0  0    0;
      0   0  0  1  1    1;
      0   0 -Zy -Yz1 0   Yz3;
      Zx1 Zx1  0  Xz1 -Xz2 Xz3;
      Yx1 -Yx1 0   0   0    0];

   b=[-R(i,1); -R(i,2); -R(i,3);
      -(R(i,3)*C(i,2)-R(i,2)*C(i,3));
      -(R(i,1)*C(i,3)-R(i,3)*C(i,1));
      -(R(i,2)*C(i,1)-R(i,1)*C(i,2))];

   Forces(:,i)=A\b;

   %Returns a vector containing Fx1,Fx2,Fy,Fz1,Fz2,Fz3
end

F=Forces(:,1)+Forces(:,2)+Forces(:,3);
```

Appendix E

Global Dyno



Coordinates for each dyno.

$Z_1 \rightarrow x_{z_1} = 0.1524\,m$
$y_{z_1} = 0.2640\,m$
$z_{z_1} = 0.1968\,m$

$Z_2 \rightarrow X_{z2} = 0.3048$
$Y_{z2} = 0$
$Z_{z2} = 0.1968$

$Z_3 \rightarrow X_{z3} = -0.1524$
$Y_{z3} = 0.2640$
$Z_{z3} = 0.1968$

$X_1 \rightarrow X_{x_1} = 0.3134$
$Y_{x1} = -0.2640$
$Z_{x1} = 0.0863$

$X_2 \rightarrow X_{x2} = 0.3134$
$Y_{x2} = 0.2640$
$Z_{x2} = 0.0863$

$Y_1 \rightarrow X_y = 0$
$Y_y = -0.2640$
$Z_y = 0.0863$

Static equilibrium $\longrightarrow$ for ForceFinder.m

$$\Sigma \vec{F}_m + \vec{R} = 0$$
$$\Sigma \vec{r}_m \times \vec{F}_m + \vec{c} \times \vec{R} = 0$$

$$\underbrace{\begin{aligned} \vec{F}_{z_1} &= F_{z_1}\hat{k} \\ \vec{F}_{z2} &= F_{z2}\hat{k} \\ \vec{F}_{z3} &= F_{z3}\hat{k} \\ \vec{F}_y &= F_y\hat{j} \\ \vec{F}_{x1} &= F_{x1}\hat{i} \\ \vec{F}_{x2} &= F_{x2}\hat{i} \end{aligned}}_{\vec{F}_m} \qquad \underbrace{\begin{aligned} \vec{r}_{z1} &= -X_{z_1}\hat{i} - Y_{z1}\hat{j} + Z_{z_1}\hat{k} \\ \vec{r}_{z2} &= X_{z2}\hat{i} + Z_{z2}\hat{k} \\ \vec{r}_{z3} &= -X_{z3}\hat{i} + Y_{z3}\hat{j} + Z_{z3}\hat{k} \\ \vec{r}_y &= -y_y\hat{j} + z_y\hat{k} \\ \vec{r}_{x1} &= X_{x1}\hat{i} - Y_{x1}\hat{j} + Z_{x1}\hat{k} \\ \vec{r}_{x2} &= X_{x2}\hat{i} + Y_{x2}\hat{j} + Z_{x2}\hat{k} \end{aligned}}_{\vec{r}_m}$$

$$\sum \vec{F}_m = -\vec{R} \rightarrow \quad F_{x_1} + F_{x_2} = -R_x$$

$$F_y = -R_y$$

$$F_{z_1} + F_{z_2} + F_{z_3} = -R_z$$

$$\sum \vec{r}_m \times \vec{F}_m + \vec{c} \times \vec{R} = 0 \rightarrow \quad -F_{z_1} Y_{z_1} + F_{z_3} Y_{z_3} - F_y z_y = -(R_z y_c - R_y z_c)$$

$$F_{z_1} X_{z_1} - F_{z_2} X_{z_2} + F_{z_3} X_{z_3} + F_{x_1} Z_{x_1} + F_{x_2} Z_{x_2} = -(R_x z_c - R_z x_c)$$

$$F_{x_1} Y_{x_1} - F_{x_2} Y_{x_2} = -(R_y x_c - R_x y_c)$$

$$\left[\begin{array}{cccccc|c}
1 & 1 & 0 & 0 & 0 & 0 & -R_x \\
0 & 0 & 1 & 0 & 0 & 0 & -R_y \\
0 & 0 & 0 & 1 & 1 & 1 & -R_z \\
0 & 0 & -z_y & -Y_{z_1} & 0 & Y_{z_3} & -(R_z y_c - R_y z_c) \\
Z_{x_1} & Z_{x_2} & 0 & X_{z_1} & -X_{z_2} & X_{z_3} & -(R_x z_c - R_z x_c) \\
Y_{x_1} & -Y_{x_2} & 0 & 0 & 0 & 0 & -(R_y x_c - R_x y_c)
\end{array}\right]$$

Appendix F

```
%Calculates the resultant force and application point knowing the forces on each dyno
%function OC=Resultant(F)
%F= mtx with the forces on each dyno
clear all
clc
%*********************************************************************
%Forces applied during calibration, 12 setups
GlobalF=[ 1000 -300 -200
          2500 -300 -200
          1000 -250 -200
         -1500 -400 -200
         -1500 200 -200
         -800  250 -200
         -1300 -750 -200
         -1000 -1000 -200
         -750 -1300 -200
         1300 750 -200
         -750 1300 -200
         1000 1000 -200]*4.45;%  forces are in Newtons
%Points of application of forces, 12 setups
C(:,:,1)=[0 0 -.3056; .4373 0 -.4556; 0 0 -.4556];
C(:,:,2)=[0 0 -.3056; 0 .4373 -.4556; 0 .4373 -.6056];
C(:,:,3)=[.4373 0 -.4556; -.2 0 -.4556; .2 0 -.4556];
C(:,:,4)=[0 0 -.3056;.2 0 -.4556; .4373 .15 -.4556];
C(:,:,5)=[0 0 -.4556; -.15 .4373 -.4556; .15 .4373 -.4556];
C(:,:,6)=[0 0 -.4556; -.15 .4373 -.4556; 0 .2 -.4556];
C(:,:,7)=[0 0 -.3065; 0 .4373 -.4556; .15 .4373 -.4556];
C(:,:,8)=[0 0 -.4556; 0 .4373 -.4556; .15 .4373 -.4556];
C(:,:,9)=[0 -.2 -.4556; 0 .4373 -.4556; .15 .4373 -.4556];
C(:,:,10)=[.4373 0 -.4556; -.2 0 -.4556; .4373 .15 -.4556];
C(:,:,11)=[0 0 -.3056; -.2 0 -.4556; .2 0 -.4556];
C(:,:,12)=[0 0 -.3056 ; .2 0 -.4556; .4373 .15 -.4556];
%*********************************************************************
figure; hold on

for ito=1:12
   F =  ForceFinder(GlobalF(ito,:),C(:,:,ito));
   %Constants
   Xz1=0.1524;
   Yz1=0.2640;
   Zz1=0.1968;
   Xz2=0.3048;
   Yz2=0;
   Zz2=Zz1;
   Xz3=Xz1;
   Yz3=Yz1;
   Zz3=Zz1;
   Xx1=0.3134;
   Yx1=.2640;
   Zx1=0.0863;
   Xy=0;
   Yy=0.2640;
   Zy=0.0863;

   R=[F(1)+F(2);
      F(3);
```

```
    F(4)+F(5)+F(6)];

  Mo=[-F(4)*Yz1+F(6)*Yz3-F(3)*Zy;
    F(4)*Xz1-F(5)*Xz2+F(6)*Xz3+F(1)*Zx1+F(2)*Zx1;
    F(1)*Yx1-F(2)*Yx1];

  MagR=sqrt(R(1)^2 + R(2)^2 + R(3)^2);
  RMo=cross(R,Mo);
  MagRMo=sqrt(RMo(1)^2 + RMo(2)^2 + RMo(3)^2);

  u=R/MagR;
  w=RMo/MagRMo;
  v=cross(w,u);

  Beta=dot(Mo,v)/MagR;

  %Plotting the lines
  xf=-100;

  for i=1:200
      xf=xf+1;
      OC(i,:)=(xf*u + Beta*w)';

  end

  OC;
  x=OC(:,1);
  y=OC(:,2);
  z=OC(:,3);

  plot3(x,y,z)
  axis square; grid on
  xlabel('X Axis')
  ylabel('Y Axis')
  zlabel('Z Axis')
end
```