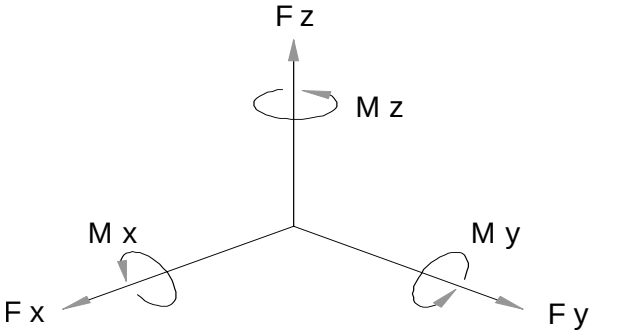
CROSS-TALK COMPENSATION USING MATRIX METHODS

David Schrand, Sensor Developments Inc.

Force sensors are designed to measure forces and torques along defined axes, typically labeled X, Y and Z. These force sensors can have from one to six measurement channels; three force channels (Fx, Fy and Fz) and three torque or moment channels (Mx, My and Mz). In theory, a load along any one of those measurement axes will not produce an output on any of the other measurement channels. Unfortunately, this is seldom the case in the real world. For most force sensors, this un-desired output, or cross talk, will be between 1 and 5%. While 1% - 5% cross talk may not sound like much, if each channel has 1% - 5% cross talk due to each of the remaining five loads, then the total cross talk could be as high as 5% - 25%.



MEASUREMENT COORDINATE SYSTEM

There are basically two methods used to reduce this potential source of measurement error. With the first method, the load-cell is 'tuned', either mechanically or electrically to reduce the channels output due to off-axis or extraneous loads. While effective, this method is time consuming and is not practical if more than two extraneous loads need to be compensated for. The other method of cross-talk compensation involves mathematically manipulating the load-cells output data to correct the cross-talk outputs. It is effective for any number of extraneous loads, and can be characterized as mathematical cross-talk compensation by the application of "cross coupling coefficients", or the inverse matrix method. This is the method that will be discussed here.

When a load is applied to a force sensor, the measurement channel that lines up with that load will respond. However, as described earlier, other measurement channels that are not in line with that applied load will also respond to that load. That's the bad news. The good news is that that response is repeatable for any given load or combination of loads. This means that by carefully applying these extraneous loads during the calibration process, and recording each channels output response to those loads, an output profile of the sensor can be created. From here, a series of simultaneous equations can be created to describe the cross talk performance of the force sensor. By solving this series of equations using any set of simultaneous data from all the channels of the sensor, the true loading condition that produced that unique set of data can be determined. The drawback to this method is that a sensor channel is required for each extraneous load that is present during loading. This usually isn't a problem, since in most instances, a measurement channel is present to monitor all the significant loads present in the application.

In application, a six axis sensor, with three force channels and three torque channels, will be calibrated by applying a known load in line with one of those measurement axis, and recording the output from the corresponding channel. This gives rise to the first transfer function for the sensor. That is, the channel output, O, is equal to the sensitivity (mV/V per unit load), K, times the applied load, F, or:

$$O_{Fx} = K_1 \cdot Fx$$

At the same time, the output from the other five channels will be recorded as cross talk outputs. Now the transfer functions for the sensor begin to take shape.

$$O_{Fx} = K_1 \cdot Fx$$
$$O_{Fy} = K_7 \cdot Fx$$
$$O_{Fz} = K_{13} \cdot Fx$$
$$O_{Mx} = K_{19} \cdot Fx$$
$$O_{My} = K_{25} \cdot Fx$$
$$O_{Mz} = K_{31} \cdot Fx$$

From this set of equations, the transfer functions (K_1, K_7, etc), can be determined for each axis by dividing the sensors output by the applied load.

This same procedure will be used to calibrate the remaining five axes of the sensor, providing the remaining transfer functions. Using the theory of superposition, they can be combined to yield the output equations that fully describe the output of the sensor to all the applied loads.

$$O_{Fx} = K_1 \cdot Fx + K_2 \cdot Fy + K_3 \cdot Fz + K_4 \cdot Mx + K_5 \cdot My + K_6 \cdot Mz$$
$$O_{Fy} = K_7 \cdot Fx + K_8 \cdot Fy + K_9 \cdot Fz + K_{10} \cdot Mx + K_{11} \cdot My + K_{12} \cdot Mz$$
$$O_{Fz} = K_{13} \cdot Fx + K_{14} \cdot Fy + K_{15} \cdot Fz + K_{16} \cdot Mx + K_{17} \cdot My + K_{18} \cdot Mz$$
$$O_{Mx} = K_{19} \cdot Fx + K_{20} \cdot Fy + K_{21} \cdot Fz + K_{22} \cdot Mx + K_{23} \cdot My + K_{24} \cdot Mz$$
$$O_{My} = K_{25} \cdot Fx + K_{26} \cdot Fy + K_{27} \cdot Fz + K_{28} \cdot Mx + K_{29} \cdot My + K_{30} \cdot Mz$$
$$O_{Mz} = K_{31} \cdot Fx + K_{32} \cdot Fy + K_{33} \cdot Fz + K_{34} \cdot Mx + K_{35} \cdot My + K_{36} \cdot Mz$$

These equations describe the output of the sensor in terms of the applied loads. However, in application, the loads are the unknowns, and the outputs are the known measured quantities. With these six equations and their six unknowns (the loads Fx-z and Mx-z), it will be possible to solve for the unknown loads.

Inverse Matrix Method

The series of equations described above can be solved by using what is called the inverse matrix method. A brief overview of the theory behind this technique is given in Appendix A, 'Inverse Matrix Theory'. In essence, it is a technique that 'inverts' the equations so that instead of having the output as a function of the loads as described above, the loads are now functions of the outputs.

Finding the inverse matrix used to correct or compensate a sensors output due to cross talk errors, involves gathering calibration data on the sensors response to extraneous loads, and using it to construct a matrix that can be used to find the sensors true loading condition. A multi-axis sensor that has undergone a thorough calibration, will already have such a cross talk matrix supplied with the calibration data. More information on the steps involved in creating this matrix is presented in Appendix B, 'Finding the Inverse Matrix'.

This may all seem confusing, but it is a very simple way to handle a very complex problem. For example, we have a sensor that is 'less than perfect'. We do however have the cross talk matrix that was supplied with the sensor, as listed below.

$$K^{-1} = \begin{vmatrix} 2.5465 & -0.0338 & 0.0575 & -0.0122 & -0.1517 & 0.0359 \\ 0.0763 & 2.4399 & -0.0336 & -0.0392 & 0.0482 & 0.0428 \\ -0.0489 & 0.0976 & 2.3759 & -0.062 & 0.021 & -0.0512 \\ -0.0666 & -0.0819 & -0.1331 & 5.5723 & -0.1013 & 0.1528 \\ -0.0255 & -0.0485 & 0.1957 & -0.1432 & 4.9941 & -0.0089 \\ 0.1941 & -0.0279 & -0.118 & -0.0969 & 0.1386 & 5.5279 \end{vmatrix}$$

With this inverse matrix the loads that produced any set of simultaneous data from the sensor can be calculated. We start with the inverse matrix, K^{-1}, and the equation:

$$X = K^{-1} \cdot O$$

Or, in long hand:

$$\begin{vmatrix} Fx \\ Fy \\ Fz \\ Mx \\ My \\ Mz \end{vmatrix} = \begin{vmatrix} 2.5465 & -0.0338 & 0.0575 & -0.0122 & -0.1517 & 0.0359 \\ 0.0763 & 2.4399 & -0.0336 & -0.0392 & 0.0482 & 0.0428 \\ -0.0489 & 0.0976 & 2.3759 & -0.062 & 0.021 & -0.0512 \\ -0.0666 & -0.0819 & -0.1331 & 5.5723 & -0.1013 & 0.1528 \\ -0.0255 & -0.0485 & 0.1957 & -0.1432 & 4.9941 & -0.0089 \\ 0.1941 & -0.0279 & -0.118 & -0.0969 & 0.1386 & 5.5279 \end{vmatrix} \cdot \begin{vmatrix} O_{Fx} \\ O_{Fy} \\ O_{Fz} \\ O_{Mx} \\ O_{My} \\ O_{Mz} \end{vmatrix}$$

At a particular point in time, we obtain the following readings from our sensor:

O_{Fx} = -1.6510 mV/V	O_{Mx} = 1.0054 mV/V
O_{Fy} = 0.6151 mV/V	O_{My} = 0.8402 mV/V
O_{Fz} = 0.2501 mV/V	O_{Mz} = 0.0067 mV/V

To find the Fx load that must have been present to create those six outputs, we multiply the elements of the first row of the inverse matrix by the individual outputs that were recorded, and sum them:

Fx = (2.5465 * -1.6510) + (-0.0338 * 0.6151) + … + (0.0359 * 0.0067) = -4.35 lb

Similarly, the remaining five loads can be calculated by carrying out the same operations on the remaining five rows of the inverse matrix. Doing so yields the following loading conditions:

Fx = -4.35 lb	Mx = 5.54 in lb
Fy = 1.37 lb	My = 4.11 in lb
Fz = 0.69 lb	Mz = -0.31 in lb

In actual use, these calculations can be set up to be done in real time by the data acquisition or control system, or can be done by post processing the data after the test is complete by using a spread sheet program. This technique will typically reduce the cross talk to less that 0.5% of the full scale capacities of the sensor.

APPENDIX A

INVERSE MATRIX THEORY

For any series of 'N' equations, each with 'N' variables, taking the form:

$$O_1 = K_1 \cdot X_1 + K_2 \cdot X_2 + \ldots K_N \cdot X_N$$
$$\ldots$$
$$O_N = K_i \cdot X_1 + K_j \cdot X_2 + \ldots K_m \cdot X_N$$

there is a unique solution to the variables X_1 thru X_N.

While this series of equations could be solved by using the substitution method from basic algebra, linear algebra's matrix math offers a simpler method. If we put this series of equations into matrix notation, we have:

$$O = K \cdot X$$

where:

$$O = \begin{vmatrix} O_1 \\ \ldots \\ O_N \end{vmatrix}$$

$$K = \begin{vmatrix} K_1 \ldots K_N \\ \ldots \\ K_i \ldots K_m \end{vmatrix}$$

$$X = \begin{vmatrix} X_1 \\ \ldots \\ X_N \end{vmatrix}$$

While we all know from basic algebra that the equation $a \cdot x = b$ can be solved for 'x' by multiplying both sides of the equation by the reciprocal or inverse of 'a' (1/a or a^{-1}), yielding the solution $x = a^{-1} \cdot b$, the same can be said for the matrix equation above. That is, the equation:

$$O = K \cdot X$$

when solved for 'X' takes the form:

$$X = K^{-1} \cdot O$$

The question now is, what is the matrix K^{-1}?

K^{-1} is the inverse of the matrix 'K'. It is not simply the matrix K with the inverse or reciprocal of all its original elements. It can be calculated by hand through a complicated series of operations on the individual rows of the matrix. Fortunately it can be done much faster with a computer. When this inverse matrix is multiplied by the solution set, here the values in the matrix 'O', the result is the unique solution to the variable matrix 'X'.

APPENDIX B

FINDING THE INVERSE MATRIX

In order to find the inverse matrix that will be used to compensate for cross talks due to extraneous loads, the sensors response to those individual extraneous loads must be determined. For example, let's say we have a six axis sensor with 5 lb capacities for the force channels, and 10 in.lb capacities for the moment channels. It has been calibrated by applying those loads, one at a time, and monitoring the individual sensor channels. The following data has been collected:

RAW OUTPUTS (mV/V)

		LOAD DIRECTION					
		Fx (5#)	Fy (5#)	Fz (5#)	Mx (10in#)	My (10in#)	Mz (10in#)
OUTPUT (mV/V)	Fx	1.9633	0.0305	-0.0524	0.0105	0.1201	-0.0261
	Fy	-0.0618	2.0473	0.0339	0.0287	-0.0439	0.0323
	Fz	0.0421	-0.0828	2.105	0.0461	-0.0136	0.0359
	Mx	0.0256	0.0287	0.0473	1.7963	0.0384	-0.0486
	My	0.0084	0.0241	-0.081	0.0501	2.0041	0.0006
	Mz	-0.0681	0.0074	0.0498	0.031	-0.0543	1.81

By dividing the channel output due to an extraneous load, by the full scale output for that same channel, the sensors cross talk performance can be seen.

CROSS TALK PERFORMANCE (% Full Scale Output)

		LOAD DIRECTION					
		Fx	Fy	Fz	Mx	My	Mz
CHANNEL	Fx		1.55	-2.67	0.53	6.12	-1.33
	Fy	-3.02		1.66	1.4	-2.14	1.58
	Fz	2	-3.93		2.19	-0.65	1.71
	Mx	1.43	1.6	2.63		2.14	-2.71
	My	0.42	1.2	-4.04	2.5		0.03
	Mz	-3.76	0.41	2.75	1.71	-3	

Here, the minimum cross talk is only 0.03% of the full scale load when an Mz load is applied to the My channel. If all the values were this good, no correction would be

necessary. However, the maximum is over 6%, and the average cross talk for this sensor is around 2%. Computed cross talk compensation will definitely improve the accuracy of the data obtained from this sensor.

The data matrix, referred to as 'K' in Appendix A, is constructed by taking the cross talk data and dividing it by the applied load, to get the transfer function for that load / channel combination.

$$K = \begin{vmatrix} 0.3927 & 0.0061 & -0.0105 & 0.0011 & 0.012 & -0.0026 \\ -0.0124 & 0.4095 & 0.0068 & 0.0029 & -0.0044 & 0.0032 \\ 0.0084 & -0.0166 & 0.421 & 0.0046 & -0.0014 & 0.0036 \\ 0.0051 & 0.0057 & 0.0095 & 0.1796 & 0.0038 & -0.0049 \\ 0.0017 & 0.0048 & -0.0162 & 0.005 & 0.2004 & 0.0001 \\ -0.0136 & 0.0015 & 0.01 & 0.0031 & -0.0054 & 0.181 \end{vmatrix}$$

Using a computer program such as Excel, the inverse of this matrix is easily calculated using built in functions, and is shown below:

$$K^{-1} = \begin{vmatrix} 2.5465 & -0.0338 & 0.0575 & -0.0122 & -0.1517 & 0.0359 \\ 0.0763 & 2.4399 & -0.0336 & -0.0392 & 0.0482 & 0.0428 \\ -0.0489 & 0.0976 & 2.3759 & -0.062 & 0.021 & -0.0512 \\ -0.0666 & -0.0819 & -0.1331 & 5.5723 & -0.1013 & 0.1528 \\ -0.0255 & -0.0485 & 0.1957 & -0.1432 & 4.9941 & -0.0089 \\ 0.1941 & -0.0279 & -0.118 & -0.0969 & 0.1386 & 5.5279 \end{vmatrix}$$

An interesting thing about matrices and their inverses is that when they are multiplied together, either as $K \cdot K^{-1}$ or as $K^{-1} \cdot K$, the result is what is called an identity matrix, that is, a matrix with 1's along its diagonal, and 0's in all the other locations, like this

$$K \cdot K^{-1} = \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{vmatrix}$$

Performing this operation is often done to verify the inverse matrix was correctly derived.