CHAPTER 18

BALANCED REDUCTION

18.1 Introduction

In this chapter another method of reducing models, "balanced reduction," will be introduced. We will compare it with the dc and peak gain ranking methods using the disk drive actuator/suspension model from the last chapter.

We have developed a strong mental picture of ranking individual modes using dc and peak gains. Furthermore, we have developed the ranking method intuitively by graphically showing how the individual modes combine to create the overall frequency response.

The concepts of controllability and observability, commonly referenced in the control community, can be used to rank modes but there is some ambiguity involved. In general, the controllability of a given mode is not related to its observability, and vice versa. The balanced reduction technique simultaneously takes into account both controllability and observability in its ranking and overcomes the uncertainty involved in using either controllability or observability alone.

We will see that for the SISO actuator model introduced in the previous chapter the balanced method provides slightly better impulse response results than the dc gain method, for models with the same number of retained modes/states. For frequency response, the balanced method fits one additional mode over that of the dc gain method, in cases where the same number of reduced modes are used for both methods.

One issue with balanced reduction is that we lose the ability to directly identify individual modes in the reduced system model. After balanced reduction one needs to examine the system matrix to identify which modes are included, while the dc and peak gain ranking techniques retain the identities of the individual modes.

Unlike SISO models, which can be easily ranked using simple dc and peak gain techniques, MIMO models will require the balanced reduction method because it easily handles the problem of ranking multiple inputs and outputs. In the next chapter we will examine a MIMO example, a disk drive actuator with a second stage of actuation in addition to the voice coil motor.

Gawronski [1996, 1998] are two excellent advanced level texts that cover balanced reduction and balanced control of structures for those interested in examining the subject more deeply.

18.2 Reviewing dc Gain Ranking, MATLAB Code balred.m

So far we have used dc or peak gains of the individual modes to rank the importance of including each mode in the reduced system. Repeating (17.1) and (17.2), the dc gain and peak gain expressions:

$$\frac{z_{ji}}{F_{ki}} = \frac{z_{nji} z_{nki}}{\omega_i^2}, \qquad (18.1)$$

$$\frac{z_{ji}}{F_{ki}} = \frac{-j}{2\zeta_i}(\text{dc gain}), \qquad (18.2)$$

where $z_{nji} z_{nki}$ is the product of the jth (output) row and kth (force applied) row terms of the ith eigenvector divided by the square of the eigenvalue for the ith mode.

For any mode, if the degree of freedom associated with the applied force has a zero value, then the force applied at that degree of freedom cannot excite that mode, so the dc and peak gains will also be zero. If the mode cannot be excited, then it has no effect on the frequency response and can be eliminated. Similarly, if the degree of freedom associated with the output has a zero value, then no matter how much force is applied to that mode, there will be no output. The dc and peak gains are zero, and the mode can be eliminated because it also will have no effect on the frequency response.

Loosely speaking, a mode which cannot be excited by the applied force is uncontrollable and a mode which has no output in the desired direction is unobservable. Conversely, modes which have "large" values for the forcing function degree of freedom are said to be "controllable" and modes with "large" values for the output degree of freedom are said to be "observable."

The code below, the input section from **balred.m**, reads in the stored output from **act8.m (**Chapter 17), stored in **act8_data.mat**. It then calculates and plots the input and output contributors to the dc gain, z_{nki} / ω_i and z_{nji} / ω_i and the resulting dc gain. This is the first time we have separated the input and output contributors to the dc gain term; in the past we have dealt only with the dc gain itself. The reason we are highlighting the two contributors is to bridge to understanding of the new concepts of controllability and observability.

```
%	balred.m	balanced modred reduction of actuator/suspension model

	clear all;

	hold off;

	clf;

	load  act8_data;

%	plot dc gain and two contibutors, force and xn, versus mode

	index_states = 1:num_modes_total-1;

	omega1 = 2*pi*freqvec';	% convert to radians

	semilogy(index_orig(2:num_modes_total)-1,gain_h0(2:num_modes_total),'k.-', ...
		index_orig(2:num_modes_total)-1,abs(force(2:num_modes_total)./ ...
		omega1(2:num_modes_total)),'k-', ...
		index_orig(2:num_modes_total)-1, ...
		abs(xn(8,2:num_modes_total)./omega1(2:num_modes_total)),'ko-')
	title([headstr ' dc gain, force and xn values versus mode number'])
	xlabel('mode number')
	ylabel('dc value')
	legend('dc gain','force','xn',3)
	grid off

	disp('execution paused to display figure, "enter" to continue'); pause
```

Figure 18.1 shows the force and output (xn) components which when multiplied create the dc gain for each mode.

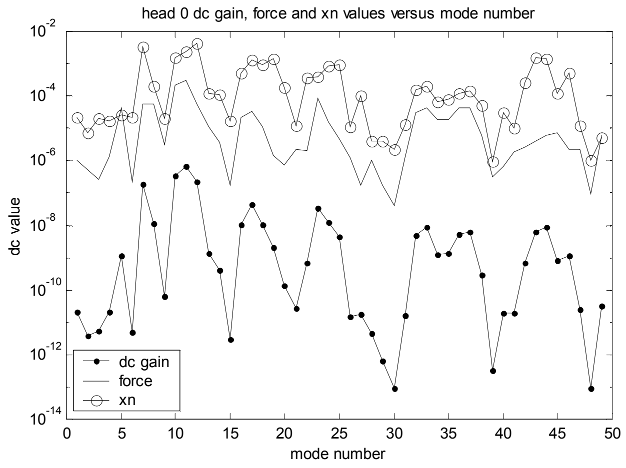

Figure 18.1: Force, output and dc gain for each mode.

It is evident from the curves of force and xn in Figure 18.1 that none of the modes has values for the input or output that go to zero, but that there is a three to four order of magnitude span for both the force and xn values. This three to four order of magnitude span for the force and xn vectors, when multiplied, results in an approximate seven order of magnitude span for the dc gain. We have used this span in dc gain values in previous chapters to rank the relative importance of modes, identifying modes for elimination.

18.3 Controllability, Observability

The intuitive descriptions of controllability and observability given above can be stated precisely using standard state space notation. See Chen [1999], Zhou [1996, 1998], Kailath [1980] and Bay [1999] for derivations and more detail.

For a state space system described by

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathrm{u} \\ \mathbf{y} &= \mathbf{C}\mathbf{x} \end{aligned} \qquad (18.3a,b)$$

the following definitions of controllability hold:

1) If there is an input "u" that can move the system from some arbitrary state $\mathbf{x}_1$ to another arbitrary state $\mathbf{x}_2$ in a finite time then the system is controllable.

2) A controllability matrix $\boldsymbol{C}$ can be formed as:

$$\boldsymbol{C} = [\mathbf{B}\ \ \mathbf{AB}\ \ \mathbf{A}^2\mathbf{B} \ldots \mathbf{A}^{n-1}\mathbf{B}] \tag{18.4}$$

If $\boldsymbol{C}$ has full (row) rank n, the system is controllable. The controllability matrix gives no insight into the relative controllability of the different modes, it shows only whether the entire system is controllable or not. If one mode of the system is not controllable, the system is not controllable.

3) Another definition of controllability involves the controllability gramian, $\mathbf{W}_c$, the solution to the Lyapunov equation:

$$\mathbf{AW}_c + \mathbf{W}_c\mathbf{A}^T + \mathbf{BB}^T = 0 \tag{18.5}$$

defined as:

$$\mathbf{W}_c = \int_0^\infty e^{\mathbf{A}\tau}\,\mathbf{BB}^T e^{\mathbf{A}^T\tau}\,d\tau \tag{18.6}$$

If the solution $\mathbf{W}_c(t)$ is non-singular (determinant is non-zero), then the system is controllable.

Diagonal elements of the controllability gramian give information about the relative controllability of the different modes and can be used in a manner similar to our use of dc gains to rank the relative controllability of individual modes.

Gramians exists only for systems that have all their poles strictly to the left of the "$j\omega$" axis. The actuator/suspension system we are analyzing has two rigid body mode poles at the origin, so we will have to analyze only the oscillatory portion of the system. We will do this by partitioning the modal form state matrices into the rigid body mode and the non-rigid body oscillatory modes. Then the definitions of controllability will be applied to only the oscillatory partition.

A similar set of definitions can be made for observability:

1) If the initial state $\mathbf{x}_o$ of a system can be inferred from knowledge of the input u and the output $\mathbf{y}$ over a finite time $(0, t)$ then the system is said to be observable.

2) An observability matrix $\boldsymbol{O}$ can be formed as:

$$\boldsymbol{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \cdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \tag{18.7}$$

If $\boldsymbol{O}$ has full (column) rank n, the system is observable.

3) Another definition of observability involves the observability gramian, $\mathbf{W}_o$, the solution to the Lyapunov equation:

$$\mathbf{A}^T\mathbf{W}_o + \mathbf{W}_o\mathbf{A} + \mathbf{C}^T\mathbf{C} = 0 \tag{18.8}$$

defined as:

$$\mathbf{W}_o = \int_0^\infty e^{\mathbf{A}^T\tau}\mathbf{C}^T\mathbf{C}e^{\mathbf{A}\tau}\,d\tau \tag{18.9}$$

If the solution $\mathbf{W}_o(t)$ is non-singular (determinant is non-zero) then the system is observable.

The diagonal elements of the observability gramian give information about the relative observability of the different modes and can be used in a manner similar to using dc gains to rank the relative observability of modes.

Because we know the form of the **A**, **B** and **C** matrices for the state space modal form, we are able to substitute those matrices into the Lyapunov equations above and derive closed form controllability and observability gramians (Gawronski 1998). It is interesting to see how the closed form gramian expressions compare with the force and xn components of the dc and peak gains. We saw earlier that the dc gain can be looked at as a product of a "force" and an "output," xn.

$$\frac{z_{ji}}{F_{ki}} = \frac{z_{nji} z_{nki}}{\omega_i^2} = \left(\frac{z_{nji}}{\omega_i}\right)\left(\frac{z_{nki}}{\omega_i}\right) = (\text{output})(\text{force}), \qquad (18.10)$$

Similarly for the peak gain at resonance:

$$\left|\frac{z_{ji}}{F_{ki}}\right| = \frac{(\text{dc gain})}{2\zeta_i} = \frac{1}{2\zeta_i}\left(\frac{z_{nji}}{\omega_i}\right)\left(\frac{z_{nki}}{\omega_i}\right) = \left(\frac{z_{nji}}{\sqrt{2\zeta_i}\,\omega_i}\right)\left(\frac{z_{nki}}{\sqrt{2\zeta_i}\,\omega_i}\right) \qquad (18.11)$$

Gawronski shows that the closed loop expression for the largest diagonal term in the 2x2 controllability gramian for mode "i" is given by:

$$w_{ci} = \frac{\|\mathbf{B}_i\|_2^2}{4\zeta_i \omega_i}, \qquad (18.12)$$

where the $\| \ \|_2$ notation represents the Euclidean norm, the square root of the sum of the squares of the elements of a vector.

The largest diagonal term in the 2x2 observability gramian for mode "i" is given by:

$$w_{oi} = \frac{\|\mathbf{C}_i\|_2^2}{4\zeta_i \omega_i} \qquad (18.13)$$

The smaller of the two diagonal terms for both the controllability and observability gramians is derived from the larger term by dividing by the square of the eigenvalue for that respective mode.

The **B** and **C** matrices for mode "i" with input at dof "k" and displacement output at dof "j" are as follows:

$$\mathbf{B}_i = \begin{bmatrix} 0 \\ F_k z_{nki} \end{bmatrix} \qquad (18.14)$$

$$\mathbf{C}_i = \begin{bmatrix} z_{nji} & 0 \end{bmatrix} \qquad (18.15)$$

Substituting into the two equations above for the closed loop gramians:

$$w_{ci} = \frac{\|\mathbf{B}_i\|_2^2}{4\zeta_i\omega_i} = \frac{\left\|\begin{bmatrix} 0 \\ F_k z_{nki} \end{bmatrix}\right\|_2^2}{4\zeta_i\omega_i} = \frac{F_k^2 z_{nki}^2}{4\zeta_i\omega_i} \tag{18.16}$$

$$w_{oi} = \frac{\|\mathbf{C}_i\|_2^2}{4\zeta_i\omega_i} = \frac{\left\|\begin{bmatrix} z_{nji} & 0 \end{bmatrix}\right\|_2^2}{4\zeta_i\omega_i} = \frac{z_{nji}^2}{4\zeta_i\omega_i} \tag{18.17}$$

Comparing the peak gain terms and the gramian terms:

Force component of dc gain: $\frac{z_{nki}}{\sqrt{2\zeta_i}\,\omega_i}$ (18.18)

Controllability diagonal: $\frac{z_{nki}^2}{4\zeta_i\omega_i}$ (18.19)

Output component of dc gain: $\frac{z_{nji}}{\sqrt{2\zeta_i}\,\omega_i}$ (18.20)

Observability diagonal: $\frac{z_{nji}^2}{4\zeta_i\omega_i}$ (18.21)

When we have ranked using peak gains, we have used the expression:

$$\text{peak gain} = \frac{z_{nji} z_{nki}}{2\zeta_i\omega_i^2} \tag{18.22}$$

If we had used the controllability and observability gramian terms for each mode to rank, we would have ranked based on

$$\frac{z_{nki}^2 z_{nji}^2}{16\zeta_i^2\omega_i^2} \tag{18.23}$$

In the controllability and observability gramian ranking of modes, we deal with the product of the squares of the eigenvector components while peak gain uses the product without squaring. Both rankings divide by the square of the eigenvalue and there is a difference in the two multipliers “2” and “16” as well as the squaring of the damping term.

18.4 Controllability, Observability Gramians

The following code section starts by defining a system which consists of the oscillatory modes of the system, excluding the first, rigid body mode. As mentioned above, gramians exist only for strictly stable systems, where all the poles strictly to the left of the "$j\omega$" axis. The two rigid body poles at the origin need to be eliminated from the system to be able to calculate gramians. In the modal form of the equations, where the modes are uncoupled, we can partition the system into rigid body and oscillatory modes. We can then calculate a reduced oscillatory system based on reducing the oscillatory modes. The full system is then ready to be re-assembled by augmenting the rigid body mode with the reduced oscillatory modes.

The controllability and observability gramians are calculated, plotted with their amplitudes on the z axis and then the diagonal entries are plotted. The position and velocity state terms are identified in each of the gramians and plotted separately.

```
%	define oscillatory system from unsorted model from act8.m, which only
%	has one output, either head 0 or head 1 so that when use balreal, will only
%	be taking into account a siso system, not the outputs of both heads 0 and 1

%	in act8.m, used output matrix with two rows so both head 0 and head 1 were available

	a_syso = a(3:asize,3:asize);	% ao is a for oscillatory system

	b_syso = b(3:asize);

	c_syso = c_disp(index_out+6,3:asize);

	syso = ss(a_syso,b_syso,c_syso,d);

%	define controllability and observability gramians for oscillatory system, syso

	wc = gram(syso,'c');

	wo = gram(syso,'o');

	[row_syso,col_syso] = size(a_syso);

	statevec = 1:row_syso;

%	calculate closed form gramians

%	define frequencies for oscillatory states

	omega1 = 2*pi*freqvec';	%  convert to radians
```

```
        ctr = 0;

        for  cnt = 1:num_modes_total

                ctr = ctr + 2;

                omega12(ctr-1) = omega1(cnt);

                omega12(ctr) = omega1(cnt);

                zeta_unsort12(ctr-1) = zeta_unsort(cnt);

                zeta_unsort12(ctr) = zeta_unsort(cnt);

        end

%       the notation below is “wc” or “wo” for controllability or observability gramians,
%       “cf” for closed-form, and “1” or “2” for maximum and minimum values for a mode

        wccf1 = (b_syso.*b_syso)./(4*zeta_unsort12(3:2*num_modes_total)' …
                      .*omega12(3:2*num_modes_total)');     % maximum terms

        wccf12 = wccf1(2:2:row_syso);                    % pick out velocity terms

        wccf2 = (b_syso.*b_syso)./(4*zeta_unsort12(3:2*num_modes_total)' …
                   .*omega12(3:2*num_modes_total)'.^3);        % minimum terms

        wccf22 = wccf2(2:2:row_syso);                    % pick out displacement terms

        wocf1 = (c_syso.*c_syso)./(4*zeta_unsort12(3:2*num_modes_total) …
                      .*omega12(3:2*num_modes_total));      % maximum terms

        wocf12 = wocf1(1:2:row_syso);                    % pick out displacement terms

        wocf2 = (c_syso.*c_syso)./(4*zeta_unsort12(3:2*num_modes_total) …
                  .*omega12(3:2*num_modes_total).^3);           % minimum terms

        wocf22 = wocf2(1:2:row_syso);                    % pick out velocity terms

%       plot controllability and observability gramians

        meshz(wc);
        view(60,30);
        title([headstr ', controllability gramian for oscillatory system'])
        xlabel('state')
        ylabel('state')
        grid on

        disp('execution paused to display figure, "enter" to continue'); pause

        meshz(wo);
        view(60,30);
        title([headstr ', observability gramian for oscillatory system'])
        xlabel('state')
        ylabel('state')
```

```
        grid on

        disp('execution paused to display figure, "enter" to continue'); pause

%       pull out diagonal elements

        wc_diag = diag(wc);

        wo_diag = diag(wo);

        modevec = 2*(1:num_modes_total-1);

%       plot diagonal terms of controllability and observability gramians, calculated with
%       gram function and closed form

        semilogy(statevec,wc_diag,'k.-',statevec(2:2:row_syso),wccf12,'ko', ...
                 statevec(1:2:row_syso),wccf22,'ko')
        title([headstr ', controllability gramian diagonal terms'])
        xlabel('states')
        ylabel('diagonal')
        legend('calculated with gram','closed form',3)
        grid off

        disp('execution paused to display figure, "enter" to continue'); pause

        semilogy(statevec,wo_diag,'k.-',statevec(1:2:row_syso),wocf12,'ko', ...
                 statevec(2:2:row_syso),wocf22,'ko')
        title([headstr ', observability gramian diagonal terms'])
        xlabel('states')
        ylabel('diagonal')
        legend('calculated with gram','closed form',3)
        grid off

        disp('execution paused to display figure, "enter" to continue'); pause

%       position and velocity states plotted separately

        semilogy(statevec(1:2:row_syso),wc_diag(1:2:row_syso),'k.-', ...
                statevec(2:2:row_syso),wc_diag(2:2:row_syso),'k-', ...
                statevec(2:2:row_syso),wccf12,'ko', ...
                statevec(1:2:row_syso),wccf22,'ko')
        title([headstr ', controllability gramian diagonal terms'])
        xlabel('states')
        ylabel('diagonal')
        legend('position states','velocity states','closed form','closed form',3)
        grid off

        disp('execution paused to display figure, "enter" to continue'); pause

        semilogy(statevec(1:2:row_syso),wo_diag(1:2:row_syso),'k.-', ...
                statevec(2:2:row_syso),wo_diag(2:2:row_syso),'k-', ...
                statevec(1:2:row_syso),wocf12,'ko', ...
                statevec(2:2:row_syso),wocf22,'ko')
        title([headstr ', observability gramian diagonal terms'])
        xlabel('states')
```

```
ylabel('diagonal')
legend('position states','velocity states','closed form','closed form',3)
grid off

disp('execution paused to display figure, "enter" to continue'); pause

semilogy(index_states,wc_diag(2:2:row_syso),'k.-', ...
                    index_states,wo_diag(1:2:row_syso),'ko-')
title([headstr ', head 0 controllability and observability state gramians'])
xlabel('mode number')
ylabel('gramian')
legend('controllability velocity state','observability position state',3)
grid off

disp('execution paused to display figure, "enter" to continue'); pause
```

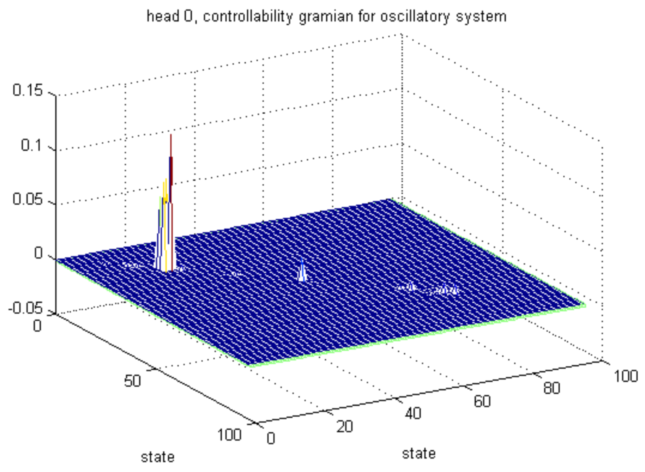

Figure 18.2: Controllability gramian values.

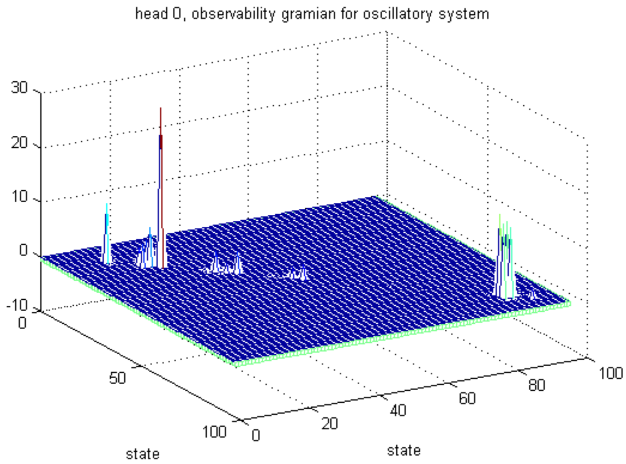

Figure 18.3: Observability gramian values.

Figures 18.2 and 18.3 plot the controllability and observability gramian values on a linear z axis scale versus location in the matrix. As noted in Gawronski [1998], for systems described in modal coordinates (with small damping, small ζ values) the gramians are diagonally dominant, meaning that the off diagonal elements are small with respect to the diagonal elements. The largest controllability terms lie along the diagonal in approximately the state 20 to 22 positions, which are the 10^{th} and 11^{th} oscillatory modes. With the rigid body mode included, these become the 11^{th} and 12^{th} modes of the full system, which we identified in the previous chapter as the two system modes in the 4 khz range and identified with the dc gain as the modes with the highest values. Note that there are not any large entries in the higher state numbers for the controllability gramian. The observability gramian plot, however, shows some very high frequency states (~80 to 100) that have circumferential motion at head 0. Intuitively, the relatively heavy coil is not going to have many modes with circumferential motion at high frequencies, while the stiff, low mass suspension will have a number of high frequency modes with circumferential motion.

The diagonal entries of both gramians are plotted versus state in Figures 18.4 and 18.5, where the odd-numbered states are position states and the even-numbered states are velocity states. Values from the "gram" function and the closed form solution (18.16) (18.17) are shown.

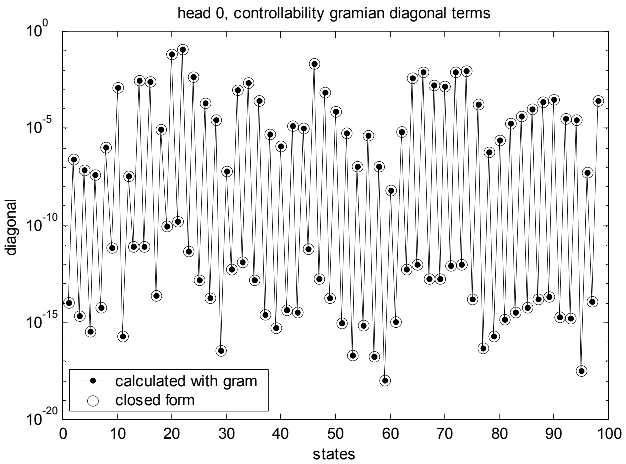

Figure 18.4: Controllability gramian diagonal terms.

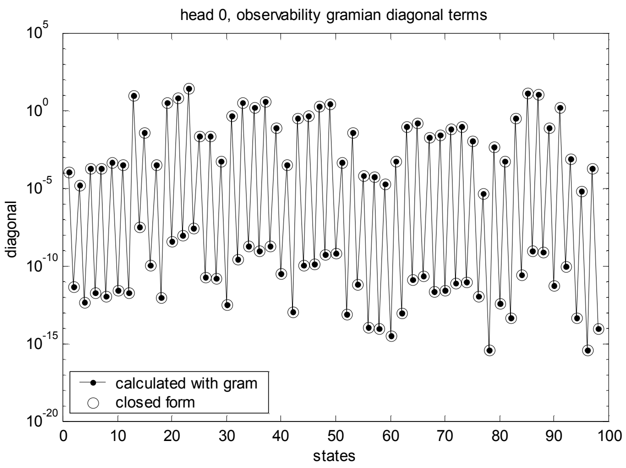

Figure 18.5: Observability gramian diagonal terms.

Figures 18.6 and 18.7 show the position and velocity terms of each gramian diagonal plotted separately. The position state and velocity state curves are offset by the square of the eigenvalue of each mode.

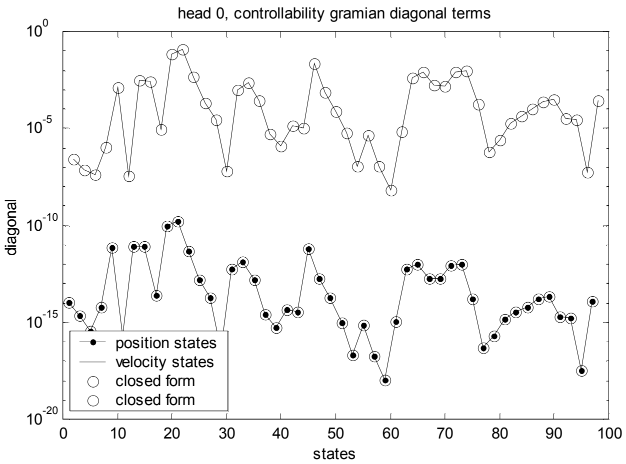

Figure 18.6: Controllability gramian diagonal position and velocity state terms.

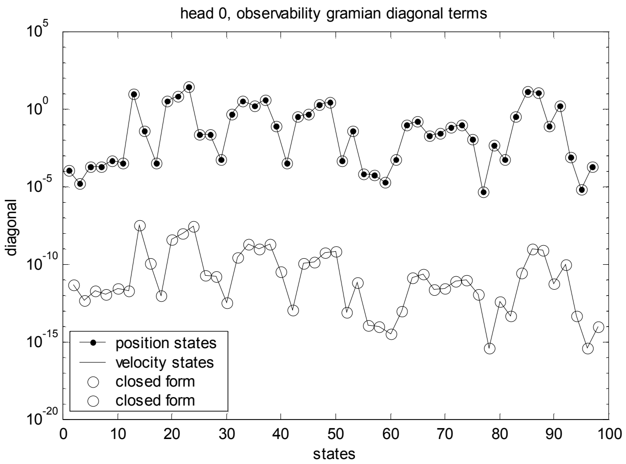

Figure 18.7: Observability gramian diagonal position and velocity state terms.

18.5 Ranking Using Controllability/Observability

Figure 18.8 shows the controllability gramian velocity state and the observability gramian position state (chosen such that the two curves have similar magnitudes for visual comparison). We could use the controllability curve to rank the states for controllability and eliminate those states with low controllability. Alternately, we could use the observability curve to rank the states for observability and then eliminate states with low observability. The problem with this approach is that the joint controllability/observability is not taken into account. There is no problem if a state chosen for elimination has a small controllability value and simultaneously a small observability value. However, if as in modes 43 and 44 (states 85 to 88) in Figure 18.8, the controllability value is small but the observability is relatively high, do we eliminate the mode or not? This is the source of ambiguity in ranking using **only** controllability or **only** observability gramians.

With the dc and peak gain ranking methods referenced earlier we used the product of the input and output (controllability measure and observability measure), jointly taking into account a measure of the controllability and observability of each mode.

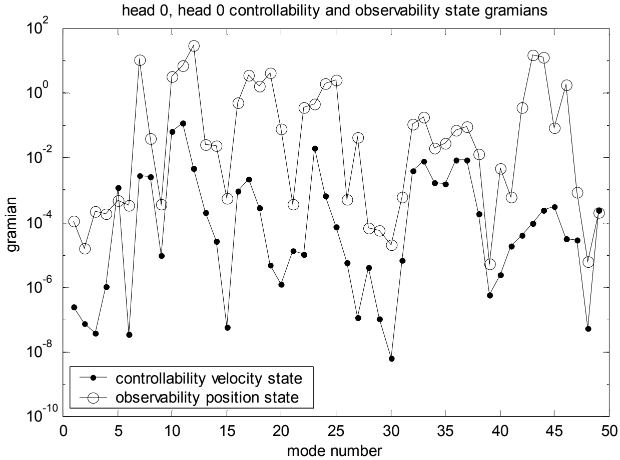

Figure 18.8: Controllability gramian velocity state and observability gramian position state diagonal terms.

18.6 Balanced Reduction

Balanced reduction was introduced in the control community by Moore [1981]. The algorithm used in the MATLAB balancing function "balreal" is taken from Laub [1987].

The algorithm creates a system with identical diagonal controllability and observability gramians. Since the two gramians are equal, **either** the diagonal or controllability gramian can be used to rank states for elimination and the ambiguity of using either only controllability or only observability is removed.

For the system "sys" defined by the following equations:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathrm{u} \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathrm{u} \end{aligned} \qquad (18.24a,b)$$

the syntax for the MATLAB "balreal" function is:

[sysb,g,T,Ti] = balreal(sys), (18.25)

where "sysb" is the new, balanced system and "g" is the diagonal of the joint gramian. "T" is the transformation matrix that is used to create "sysb." "Ti" is the inverse of "T."

The diagonal terms of the joint gramian, g, are squares of the Hankel singular values of the system. The Hankel matrix is the product of the controllability and observability gramians. Hankel singular values are the squares of the eigenvalues of the Hankel matrix. See Gawronski [1998] for a MATLAB script "bal_op_loop.m" that uses Singular Value Decomposition to calculate the Hankel singular values.

T is the state transformation matrix that is used along with its inverse, $\mathbf{T}^{-1}$, to create "sysb" from "sys" using:

$$\begin{aligned} \dot{\mathbf{x}}_\mathrm{b} &= \mathbf{T}\mathbf{A}\mathbf{T}^{-1}\mathbf{x}_\mathrm{b} + \mathbf{T}\mathbf{B}\mathrm{u} \\ \mathbf{y} &= \mathbf{C}\mathbf{T}^{-1}\mathbf{x}_\mathrm{b} + \mathbf{D}\mathrm{u} \end{aligned} \qquad (18.26a,b)$$

The gramians are also transformed by **T** to identical diagonal form:

$$\mathbf{W}_\mathrm{bo} = \mathbf{W}_\mathrm{bc} = \mathrm{diag(g)} \qquad (18.27)$$

Because the controllability and observability gramians are identical, there is no ambiguity in deciding whether the most controllable or the most observable states should be chosen. The states to be kept are the states with the largest diagonal terms.

The code below uses "balreal" to calculate the balanced system, "sysob," and plots the resulting gramians.

```
%		use balreal to rank oscillatory states and modred to reduce for comparison

		[sysob,g,T,Ti] = balreal(syso);
```

```
%           define controllability and observability gramians for balanced
%           oscillatory system, sysob

            wcb = gram(sysob,'c');

            wob = gram(sysob,'o');

            wcb_diag = diag(wcb);

            wob_diag = diag(wob);

            modevec = 2*(1:num_modes_total-1);

%           plot balanced controllability and observability gramians

            meshz(wcb);
            view(60,30);
            title([headstr ', oscillatory system balanced controllability gramian'])
            xlabel('state')
            ylabel('state')
            grid on

            disp('execution paused to display figure, "enter" to continue'); pause

            meshz(wob);
            view(60,30);
            title([headstr ', oscillatory system balanced observability gramian'])
            xlabel('state')
            ylabel('state')
            grid on

            disp('execution paused to display figure, "enter" to continue'); pause

%           plot diagonal terms of balanced controllability and observability gramians

            semilogy(statevec,wcb_diag,'k.-',statevec,wob_diag,'ko-')
            title([headstr ', balanced system controllability and observability gramian …
                          diagonal terms'])
            xlabel('states')
            ylabel('diagonal')
            legend('controllability','observability',3)
            grid off

            disp('execution paused to display figure, "enter" to continue'); pause
```

Figures 18.9 and 18.10 plot terms of the controllability and observability gramian matrices for the balanced system, with the values plotted along the z axis. Comparing them to the original, unbalanced, controllability and observability gramian plots in Figures 18.2 and 18.3, we see that the balanced plots are identical and strictly diagonal.

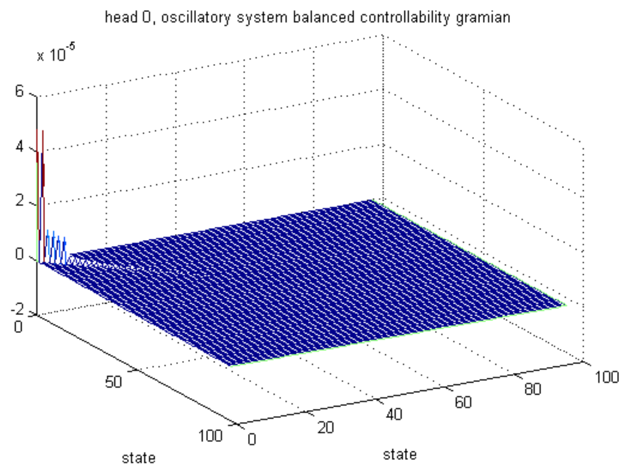

Figure 18.9: Balanced controllability gramian.

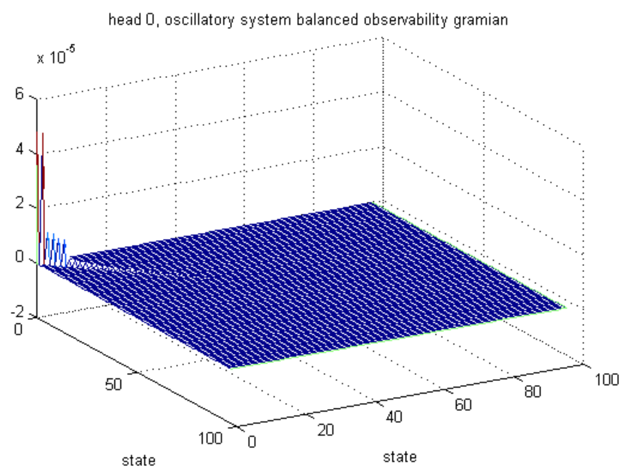

Figure 18.10: Balanced observability gramian.

Plotting diagonal terms of the controllability and observability gramians versus states, Figure 18.11, shows that the two curves overlay one another and that they are ranked from large to small by virtue of the balancing operations.

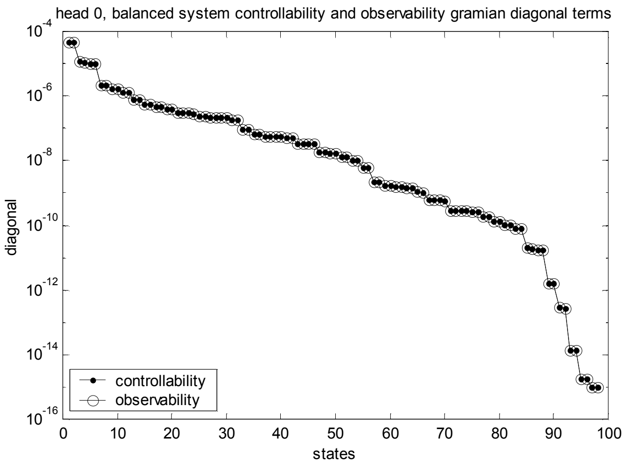

Figure 18.11: Balanced system controllability and observability gramian diagonal terms.

We are now in a position to use the balanced system gramian (either controllability or observability) to decide which states are relatively less important and can be eliminated. Since the states in the balanced system are organized from most to least significant, the MATLAB function "modred" can be used with either the "del" or "mdc" option to eliminate the states with the lowest joint controllability/observability, the higher numbered states in the balanced system.

18.7 Balanced and dc Gain Ranking Frequency Response Comparison

The code in this section starts by plotting the Hankel singular values and the sorted dc gain of the oscillatory modes to see their similarities. The modred function is then used to reduce the system to the number of modes chosen in the last **act8.m** run, using both the "del" and "mdc" options. The complete system is then rebuilt by augmenting the reduced oscillatory system with the rigid body mode. Finally, the code plots frequency responses and compares the results of dc gain ranking from **act8.m** and balanced ranking from **balred.m**.

```
%       plot sorted diagonal values and dc gain

        [row_syso,col_syso] = size(a_syso);

        semilogy(statevec,g,'k.-',2*index_orig((2:num_modes_total)-1), ...
                gain_h0_sort(2:num_modes_total),'k-')
        title([headstr ', sorted diagonal terms of balanced gramian and dc gain'])
        xlabel('state')
```

```
        ylabel('diagonal of gramian')
        legend('balanced','dc gain',3)
        grid off

        disp('execution paused to display figure, "enter" to continue'); pause

        num_oscil_states_used = 2*num_modes_used - 2;

%       use modred to reduce states from balanced system using both "del" and "mdc"

        bsys_delo = modred(sysob,num_oscil_states_used+1:2*num_modes_total-2,'del');

        bsys_mdco = modred(sysob,num_oscil_states_used+1:2*num_modes_total-2,'mdc');

%       rebuild system by appending balanced realization of oscillatory modes to
%                       rigid body mode

        [a_delo_bal,b_delo_bal,c_delo_bal,d_delo_bal] = ssdata(bsys_delo);

        a_del_bal = [   a(1:2,1:2)      zeros(2,num_oscil_states_used)
                     zeros(num_oscil_states_used,2)      a_delo_bal    ];

        b_del_bal = [b(1:2,:)
                     b_delo_bal];

        c_del_bal = [c_disp(index_out+6,1:2) c_delo_bal];

        bsys_del = ss(a_del_bal,b_del_bal,c_del_bal,d);

        [a_mdco_bal,b_mdco_bal,c_mdco_bal,d_mdco_bal] = ssdata(bsys_mdco);

        a_mdc_bal = [   a(1:2,1:2)      zeros(2,num_oscil_states_used)
                     zeros(num_oscil_states_used,2)      a_mdco_bal    ];

        b_mdc_bal = [b(1:2,:)
                     b_mdco_bal];

        c_mdc_bal = [c_disp(index_out+6,1:2) c_mdco_bal];

        bsys_mdc = ss(a_mdc_bal,b_mdc_bal,c_mdc_bal,d);

        [magr_del,phsr_del] = bode(bsys_del,frad);

        [magr_mdc,phsr_mdc] = bode(bsys_mdc,frad);

%       compare frequency responses for all four reduction methods

        loglog(f,mag(index_out,:),'k--',f,mag_sort_red(index_out,:),'k-', ...
                        f,magr_del(1,:),'k.-')
        title([headstr ', results comparison, ',num2str(num_modes_used),' modes, ', ...
                num2str(num_oscil_states_used),' oscillatory balanced states'])
        xlabel('Frequency, hz')
        ylabel('Magnitude, mm')
        axis([500 25000 1e-8 1e-4])
        legend('all modes','sorted truncated','balreal modred del',3)
```

```
grid off

disp('execution paused to display figure, "enter" to continue'); pause

loglog(f,mag(index_out,:),'k--',f,mag_mdc(index_out,:),'k-',f,magr_mdc(1,:),'k.-')
title([headstr ', results comparison, ',num2str(num_modes_used),' modes, ', ...
        num2str(num_oscil_states_used),' oscillatory balanced states'])
xlabel('Frequency, hz')
ylabel('Magnitude, mm')
axis([500 25000 1e-8 1e-4])
legend('all modes','sorted mdc','balreal modred mdc',3)
grid off

disp('execution paused to display figure, "enter" to continue'); pause
```

Figure 18.12 shows the Hankel singular values and sorted dc gains versus number of states. At this point it is interesting to compare frequency responses for the two ranking techniques to see how each decides which modes/states to eliminate.

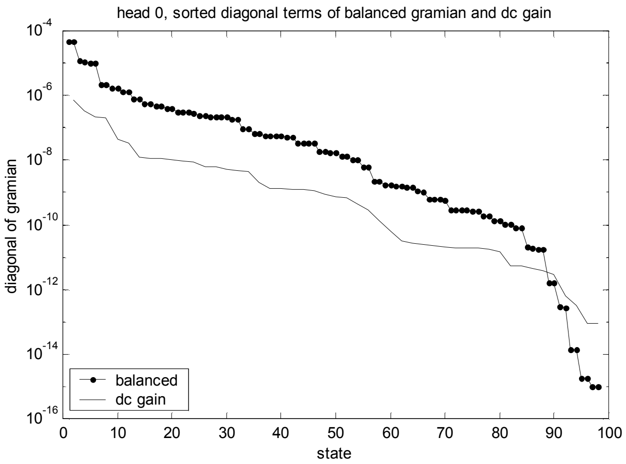

Figure 18.12: Balanced gramian diagonal terms (Hankel singular values) and sorted dc gain.

Figures 18.13 to 18.18 show frequency response plots for different numbers of retained modes, from two to seven modes, including the rigid body mode.

While the code above calculates "sorted truncated" and "balreal modred del" responses, we will only show the following in the figures below:

1) "sorted mdc" – uses dc gain ranking and modred "mdc" to reduce

2) "balreal modred mdc" – uses balreal for ranking and modred "mdc" to reduce

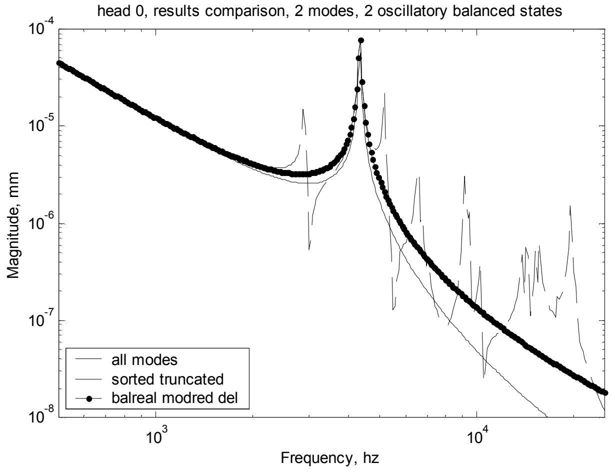

Figure 18.13: Two modes included.

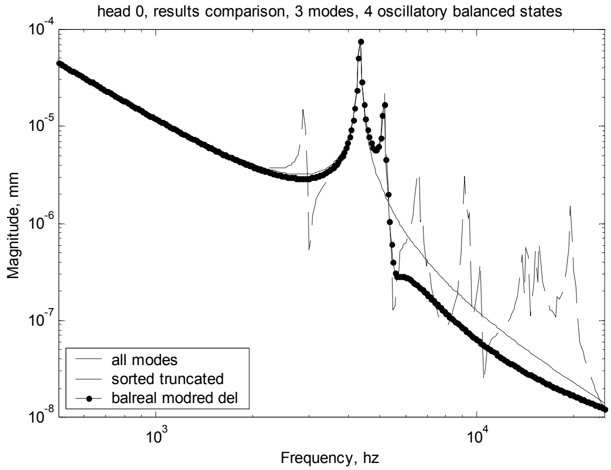

Figure 18.14: Three modes included.

Note that the two ranking methods chose different modes for the three reduced modes. The dc gain method chose the two system modes in the 4.2 khz range (almost coincident) while the balanced method chose one mode at 4.2 khz and another at 5.1 khz.

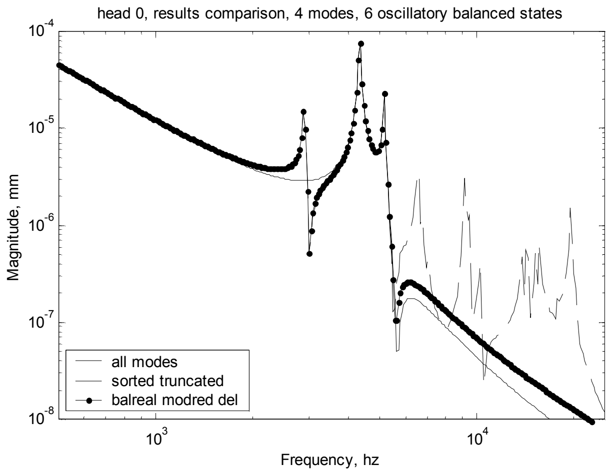

Figure 18.15: Four modes included.

For the four reduced mode case, the dc gain method picked up the 5.1 khz mode, while the balanced method chose the suspension torsion mode at 2.9 khz.

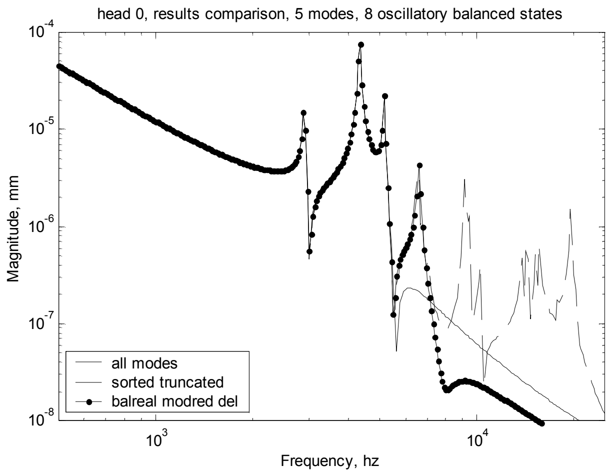

Figure 18.16: Five modes included.

For the five reduced mode case the dc gain method included the torsion mode but missed the mode at 5.5 khz which was picked up by the balanced method.

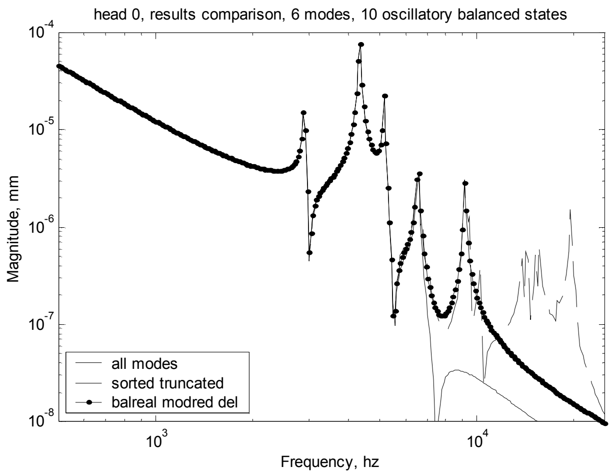

Figure 18.17: Six modes included.

With six reduced modes the balanced method includes the mode at 9 khz, but the dc gain method missed it.

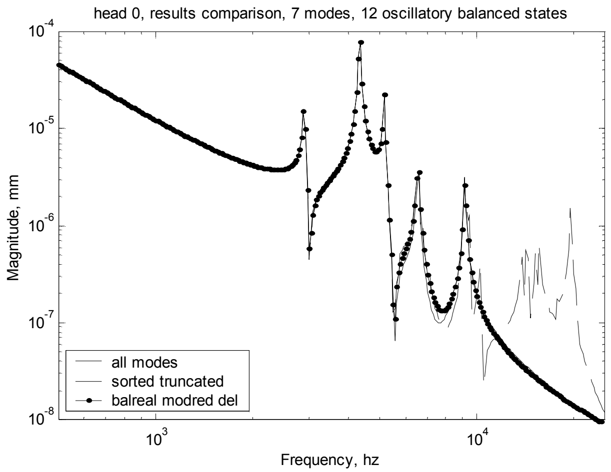

Figure 18.18: Seven modes included.

With seven or higher modes the balanced and dc gain results are very similar. We will see later when analyzing impulse responses of the oscillatory system that the two methods give results which are within a few percent of each other when seven or more modes are included in the reduced model.

18.8 Balanced and dc Gain Ranking Impulse Response Comparison

This section will compare the impulse responses for four different reduced systems, using from 2 through 15 modes. Only the matched dc gain (mdc) methods will be compared as there are minimal differences between the mdc method and the truncation or "del" method of reducing, as can be seen from the eight reduced mode results below.

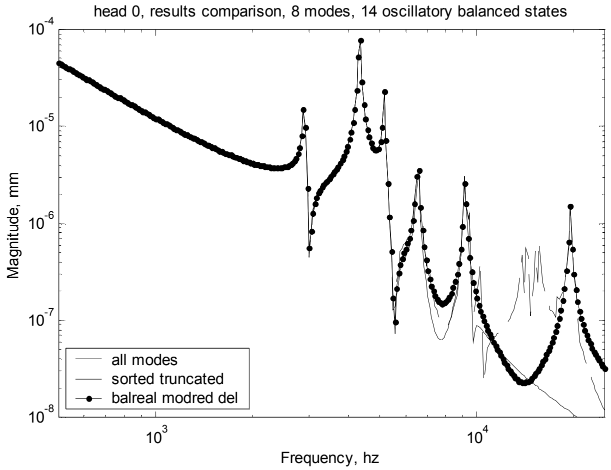

Figure 18.19: Frequency response for eight-mode reduced models, sorted truncated and balreal modred "del."

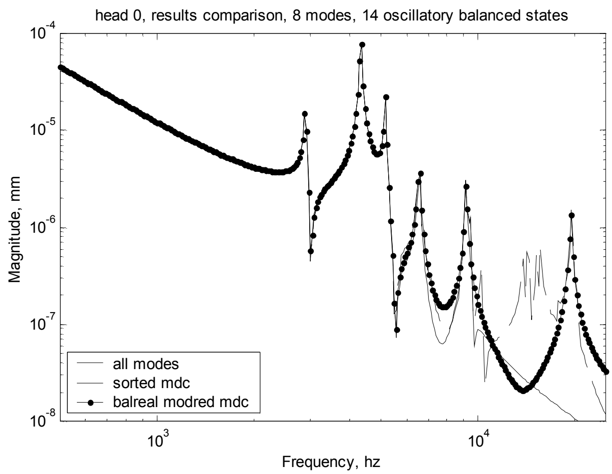

Figure 18.20: Frequency response for 8-mode reduced models, sorted "mdc" and balreal modred "mdc."

In studying the impulse response, we will use only the oscillatory modes. The final model will of course include the rigid body mode, but to study the effects of the various reduced models on transient response it is useful to include only the oscillatory modes. The reason this is useful is that a typical forcing function applied to a rigid body mode will move the system from one position to another, with rigid body displacements quite large relative to the displacements of the oscillatory modes, creating roundoff errors that mask the oscillatory mode responses.

The code below calculates the impulse response using the "lsim" function for five oscillatory systems, the original "all modes included" system and the four reduced systems. The impulse responses are then plotted and the normalized reduction index, δ (Gawronski 1998), is calculated, where the index is defined as:

$$\delta = \frac{\left\| \text{disp(all mode model)} - \text{disp(reduced model)} \right\|}{\left\| \text{disp(all mode model)} \right\|} \tag{18.28}$$

A table of results for δ from earlier runs with different numbers of retained modes is included in the code listing below. Information in the table is also shown graphically in Figures 18.25 and 18.26.

```
%        calculate impulse responses of all four oscillatory systems for comparison

         ttotal = 0.0025;
```

```
        t = linspace(0,ttotal,400);

%       define oscillatory systems for models

%       sorted reduced system

        red_size = 2*num_modes_used;

        [a_sys_sort_red,b_sys_sort_red,c_sys_sort_red,d_sys_sort_red] = …
                                ssdata(sys_sort_red);

        a_sys_sort_redo = a_sys_sort_red(3:red_size,3:red_size);

        b_sys_sort_redo = b_sys_sort_red(3:red_size);

        c_sys_sort_redo = c_sys_sort_red(index_out,3:red_size);

        sys_sort_redo = ss(a_sys_sort_redo,b_sys_sort_redo,c_sys_sort_redo,d);

%       sorted mdc reduced system

        [a_sys_sort_mdc,b_sys_sort_mdc,c_sys_sort_mdc,d_sys_sort_mdc] = …
                                ssdata(sys_mdc);

        a_sys_sort_mdc = a_sys_sort_red(3:red_size,3:red_size);

        b_sys_sort_mdc = b_sys_sort_red(3:red_size);

        c_sys_sort_mdc = c_sys_sort_red(index_out,3:red_size);

        sys_mdco = ss(a_sys_sort_mdc,b_sys_sort_mdc,c_sys_sort_mdc,d);

%       use lsim to calculate transient response

        [disp_syso,t_syso] = impulse(syso,t);

        [disp_sys_sort_redo,t_sys_sort_redo] = impulse(sys_sort_redo,t);

        [disp_sys_sort_mdco,t_sys_sort_mdco] = impulse(sys_mdco,t);

        [disp_bsys_delo,t_bsys_delo] = impulse(bsys_delo,t);

        [disp_bsys_mdco,t_bsys_mdco] = impulse(bsys_mdco,t);

%       build matrix of results

        dispo = [disp_syso(:,1) disp_sys_sort_redo(:,1) ...
                disp_sys_sort_mdco(:,1) disp_bsys_delo(:,1) ...
                disp_bsys_mdco(:,1)];

        sort_redo_del = dispo(:,1) - dispo(:,2);

        sort_mdco_del = dispo(:,1) - dispo(:,3);
```

```
            delo_del = dispo(:,1) - dispo(:,4);

            mdco_del = dispo(:,1) - dispo(:,5);

%           calculate normalized reduction index

            index_sort_redo =  …
                 sqrt(sum(sort_redo_del.*sort_redo_del))/sqrt(sum(dispo(:,1).*dispo(:,1)))

            index_sort_mdco = …
               sqrt(sum(sort_mdco_del.*sort_mdco_del))/sqrt(sum(dispo(:,1).*dispo(:,1)))

            index_delo =  …
                   sqrt(sum(delo_del.*delo_del))/sqrt(sum(dispo(:,1).*dispo(:,1)))

            index_mdco =  …
                     sqrt(sum(mdco_del.*mdco_del))/sqrt(sum(dispo(:,1).*dispo(:,1)))

            [num_modes_used index_sort_redo index_sort_mdco index_delo index_mdco]

            plot(t_syso,disp_syso(:,1),'k-',t_sys_sort_redo,disp_sys_sort_redo(:,1),'k.-')
            title([headstr ', displacement vs time, ',num2str(num_modes_used-1), …
                              ' oscillatory modes'])
            xlabel('time, sec')
            ylabel('displacement, mm')
            legend('all modes','sorted reduced system',4)
            grid off

            disp('execution paused to display figure, "enter" to continue'); pause

            plot(t_syso,disp_syso(:,1),'k-',t_sys_sort_mdco,disp_sys_sort_mdco(:,1),'k.-')
            title([headstr ', displacement vs time, ',num2str(num_modes_used-1), …
                                ' oscillatory modes'])
            xlabel('time, sec')
            ylabel('displacement, mm')
            legend('all modes','sorted modred mdc',4)
            grid off

            disp('execution paused to display figure, "enter" to continue'); pause

            plot(t_syso,disp_syso(:,1),'k-',t_bsys_delo,disp_bsys_delo(:,1),'k.-')
            title([headstr ', displacement vs time, ',num2str(num_oscil_states_used), …
                                     ' oscillatory balanced states'])
            xlabel('time, sec')
            ylabel('displacement, mm')
            legend('all modes','balreal modred del',4)
            grid off

            disp('execution paused to display figure, "enter" to continue'); pause

            plot(t_syso,disp_syso(:,1),'k-',t_bsys_mdco,disp_bsys_mdco(:,1),'k.-')
            title([headstr ', displacement vs time, ',num2str(num_oscil_states_used), …
                                     ' oscillatory balanced states'])
            xlabel('time, sec')
            ylabel('displacement, mm')
```

```
        legend('all modes','balreal modred mdc',4)
        grid off

        disp('execution paused to display figure, "enter" to continue'); pause

%       plot results of oscillatory impulse response normalized error index versus
%       number of modes used

        error_norm = [    2     .4332   .4332   0.3007   0.3008
                          3     .3041   .3041   0.1777   0.1823
                          4     .1759   .1759   0.1135   0.1137
                          5     .1134   .1134   0.0845   0.0841
                          6     .0851   .0851   0.0598   0.0603
                          7     .0637   .0637   0.0582   0.0583
                          8     .0599   .0599   0.0383   0.0401
                          9     .0594   .0594   0.0343   0.0356
                          10    .0572   .0572   0.0338   0.0347
                          11    .0555   .0555   0.0258   0.0264
                          12    .0392   .0392   0.0280   0.0268
                          13    .0327   .0327   0.0167   0.0168
                          14    .0270   .0270   0.0162   0.0158
                          15    .0209   .0209   0.0162   0.0156];

        nmode = error_norm(:,1);

        error_sort_red = error_norm(:,2);

        error_sort_mdc = error_norm(:,3);

        error_bal_del = error_norm(:,4);

        error_bal_mdc = error_norm(:,5);

        plot(nmode,error_sort_red,'k.-',nmode,error_bal_del,'ko-')
        title([headstr ', normalized reduction index versus number of modes included'])
        xlabel('number of modes included')
        ylabel('normalized reduction index')
        legend('sorted reduced','balanced del')
        axis([0 15 0 0.5])
        grid off

        disp('execution paused to display figure, "enter" to continue'); pause

        plot(nmode,error_sort_mdc,'k.-',nmode,error_bal_mdc,'ko-')
        title([headstr ', normalized reduction index versus number of modes included'])
        xlabel('number of modes included')
        ylabel('normalized reduction index')
        legend('sorted mdc','balanced mdc')
        axis([0 15 0 0.5])
        grid off

        disp('execution paused to display figure, "enter" to continue'); pause

        save  balred_data;
```

The impulse response comparisons for the same four reduced methods are shown in the four figures below.

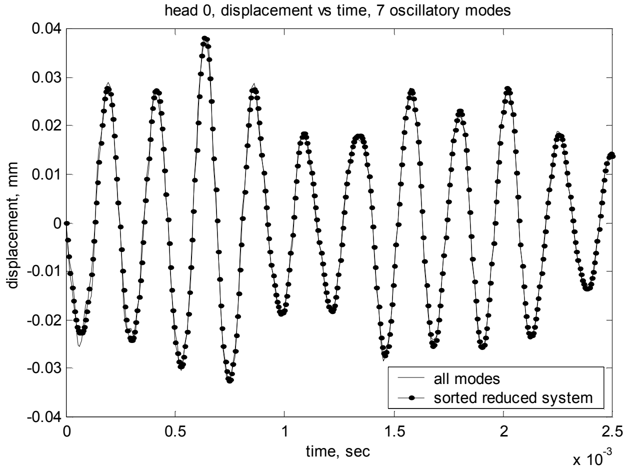

Figure 18.21: Impulse response comparisons for oscillatory system, full model (all oscillatory modes) and sorted reduced system with seven oscillatory modes.

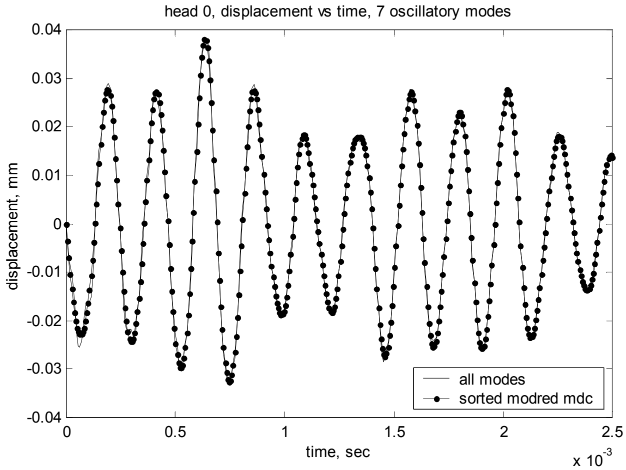

Figure 18.22: Impulse response comparisons for oscillatory system, full model (all oscillatory modes) and sorted modred with "mdc" option with seven oscillatory modes.

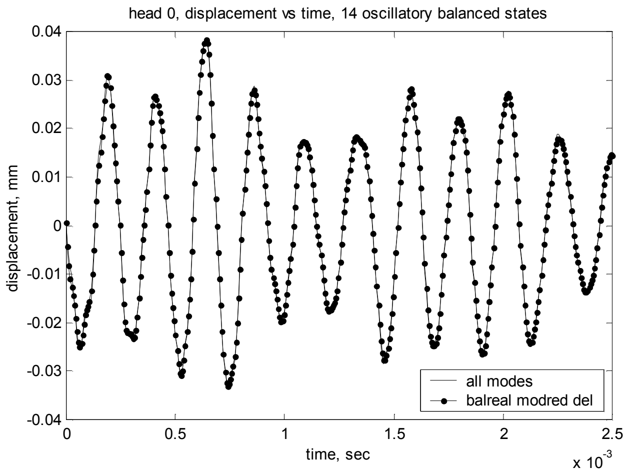

Figure 18.23: Impulse response comparisons for oscillatory system, full model (all oscillatory modes) and balreal modred "del" reduced system with seven oscillatory modes.

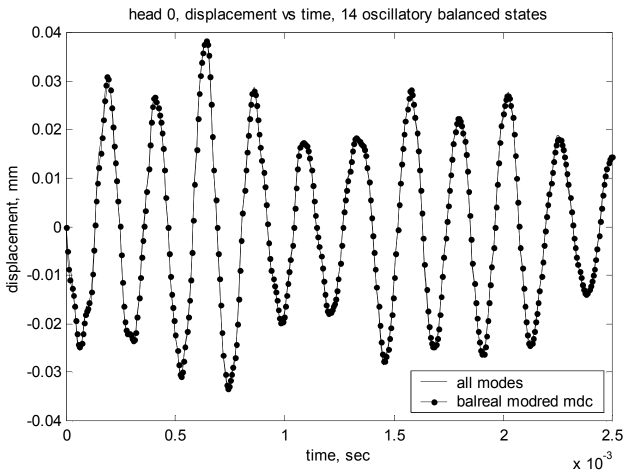

Figure 18.24: Impulse response comparisons for oscillatory system, full model (all oscillatory modes) and balreal modred "mdc" reduced system with seven oscillatory modes.

The two figures below compare the normalized reduction index, δ, as a function of the number of modes included in the various reduced model methods.

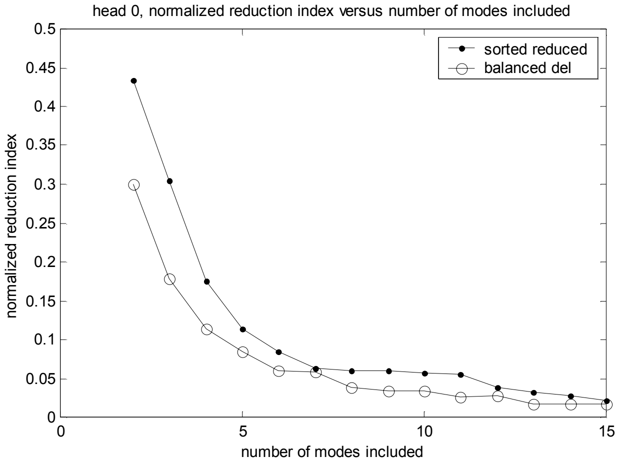

Figure 18.25: Impulse response normalized reduction index versus number of modes included in reduction for sorted reduced and balanced modred "del" option reductions.

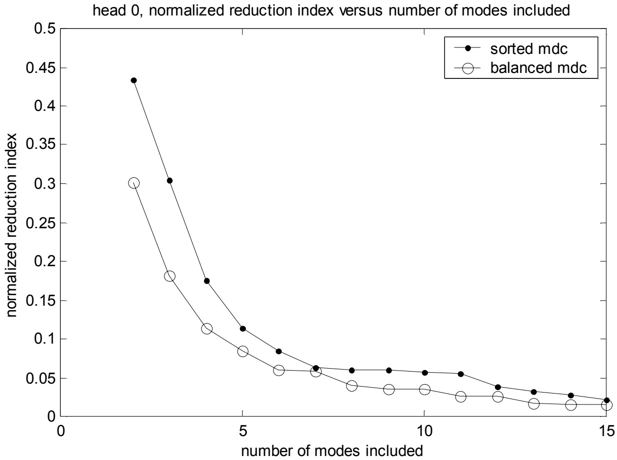

Figure 18.26: Impulse response normalized reduction index versus number of modes included in reduction for sorted modred "mdc" and balanced modred "mdc" options reductions.

As mentioned in the frequency response section, when five or more modes are included, the impulse responses are almost identical for all reduction techniques, with small differences in normalized reduction indices. For less than five modes, it is better to use the balanced technique because it picks up an additional mode in addition to the system mode, whereas the dc gain method assigns the first two modes to the almost coincident two modes near the system mode.