An Expert System for Designing Input Shapers

by

Lila S. French

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

May 22, 1999

[June 1999]

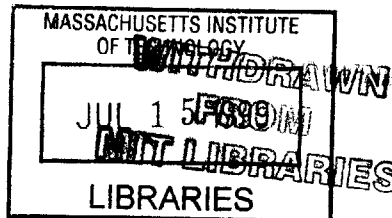© Copyright 1999 Lila S. French. All rights reserved.

The author hereby grants to M.I.T. permission to reproduce and
distribute publicly paper and electronic copies of this thesis
and to grant others the right to do so.

Author_____
    Department of Electrical Engineering and Computer Science
                                                May 22, 1999

Certified by____   _____

Accepted by_____

          Chai

An Expert System for Designing Input Shapers

by
Lila S. French

Submitted to the
Department of Electrical Engineering and Computer Science

May 22, 1999

In Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science

ABSTRACT

Many mechanical systems have parts that require quick and precise movements. This task is often difficult to accomplish, because as the parts are moved more quickly, more residual vibration is produced. Input shaping is a command generation technique that reduces this vibration by driving the system with a specially designed input command, determined by convolving a sequence of impulses, known as the input shaper, with any desired system command. Numerous shapers have been proposed, each with unique properties. The selection of an input shaper for any given system has generally relied to some degree on the knowledge of an expert. I have developed an expert system in an attempt to capture the expertise involved in input shaper design. Furthermore, the design of the input shaper can be refined if the system response is fed back into the expert system.

Thesis Supervisor: Warren P. Seering
Title: Professor / Director of the Center for Innovation in Product Development

Table of Contents

Table of Figures

Table of Tables

1. Introduction

Many mechanical systems have parts that require quick and precise movements. This task is difficult to accomplish, because the parts are usually flexible. That is, as the parts are moved more quickly, more residual vibration, and hence, less precision, results. One way mechanical engineers handle this problem is by applying a feed-forward control technique called *input shaping*.

Input shaping is a command generation technique patented by MIT in 1990, and used today by thousands of machines around the world. The idea behind input shaping is that dependent on the fundamental characteristics of the machine among other factors, we can drive it with a series of precisely timed and sized movements that cancel the residual vibration. This new command is obtained by convolving a sequence of impulses, known as the *input shaper*, with the desired system command.

Many types of input shapers have been proposed. The best input shaper for a given system depends on many factors including model inaccuracies, actuator limits, existence of unmodeled high modes, trajectory tracking requirements, rise time and settling time requirements, and computational limitations. Given a set of performance specifications, the process of choosing the best input shaper is not well defined. To date, this decision process has relied on the skill and experience of the controls engineer.

In order to facilitate the design of input shapers, I have developed an expert system. The expert system takes as input a codified description of the system dynamics and performance requirements. The input set is then filtered through a set of rules to produce the input shaper that best satisfies these requirements. The set of rules attempts

to capture the expertise of input shaper designers and the known trade-offs between the various types of input shapers.

The process of selecting an input shaper is an excellent application for an expert system. Not only can the process be represented as a combination of calculations and weighted rules, but it is also almost arbitrarily complex. The expert often has a very large number of system parameters to consider as well as a large number of input shapers from which to choose. Consequently, an expert system could benefit both the expert and non-specialized user. In effect, the expert system could provide an easily accessible database containing a large number of possible input shapers.

One issue that arises when applying a new command signal to a system is that it often produces unexpected results. The new command may excite a high frequency, the actuators may become saturated, or simply the rise time or settling time may be longer than expected. Given an unsatisfactory response, the expert can make adjustments to the system model based on the response information available. The expert can then re-design the input shaper and apply the refined version to the system. This refinement expertise is also incorporated into the expert system.

This thesis describes the implementation and testing of the expert system described. I first present the background information on input shaping necessary to demonstrate the utility of an expert system in the field. I then describe the input shaper selection part of the expert system. In Section 4, I describe the shaper refinement part of the system. I discuss why refinement is necessary and how it is implemented in the system. Finally, I discuss the experiments run on the system and the results obtained.

2. Background Information

2.1 Input Shaping

Nearly every structure in the world will vibrate when moved, especially when pushed to its performance limit. One excellent example is a crane. When a crane is moved quickly from one place to another, it usually continues to swing back and forth after the actuator input has ceased. For example, the input and response shown in Figure 1a represent a typical gantry crane subject to a step input. One way to decrease this vibration would be to move the crane more slowly, say with a ramp input like that shown in Figure 1b. The result is less vibration; however, we would like to be able to quickly perform the task with nearly zero residual vibration. This can be accomplished by specially shaped command signals like that shown in Figure 1c. To create such a command we convolve our original step command with a carefully designed series of impulses. This process is shown in Figure 2. The shaper essentially divides the step input into two separate moves. The first move causes vibration, and the second move cancels it.

Figure 2: The Input Shaping Process. [10]

Figure 1: Response of a Flexible System to Three Different Reference Commands. [12]

Another promising application for input shaping is in disk drive manufacturing. A sketch of a machine that tests disk drive reading heads is shown in Figure 3. The machine positions the reading head over a calibrated hard disk. If the head being tested can correctly read the information, then the reading head is accepted. As with the crane, the problem here is that the machine vibrates before settling to the correct location. Figure 4 shows the response of the machine with and without input shaping. The shaped output has been vertically offset to provide a clearer comparison. As the figure indicates, input shaping greatly reduces the residual vibration of the machine.

8
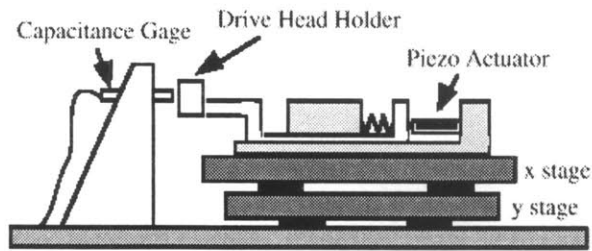
Figure 3: A Disk Drive Reading-Head Tester. [14]
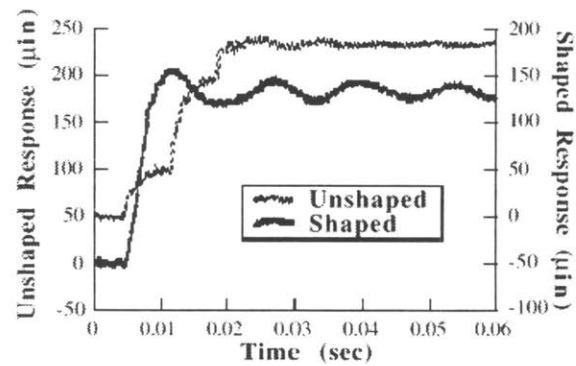
Figure 4: Shaped and Unshaped Responses of a Disk Drive Reading-Head Tester. [14]

2.2 Types of Input Shapers

Given the large number of possible applications, a great variety of input shapers have been developed. Depending on the flexible structure's characteristics and performance specifications, an expert must choose from the various input shapers. One characteristic that distinguishes each input shaper is its robustness to modeling errors. To demonstrate robustness, the residual vibration amplitude is plotted as a function of a modeling error. These sensitivity curves for two common shapers are shown in Figure 5. The robustness can be compared quantitatively by measuring the width of the curves at some low level of vibration. An input shaper with a high insensitivity allows the system to maintain low vibration levels for a large range of frequencies around the modeling frequency. The zero vibration and derivative (ZVD) shaper was the first input shaper designed to be robust to modeling errors [10]. Its name comes from its sensitivity curve shown in Figure 5. It produces zero vibration when the actual system frequency, ω_a, perfectly matches the modeling frequency, ω_m. Furthermore, the derivative of its sensitivity curve is zero at the modeling frequency.

9

One technique for widening the curve, and thereby increasing robustness, relaxes the requirement of exactly zero residual vibration when the model is perfect [12, 16]. We can see from Figure 5 that such an Extra-Insensitive (EI) shaper has a higher insensitivity than the ZVD shaper (0.40 compared to 0.29) for a 5% vibration level.
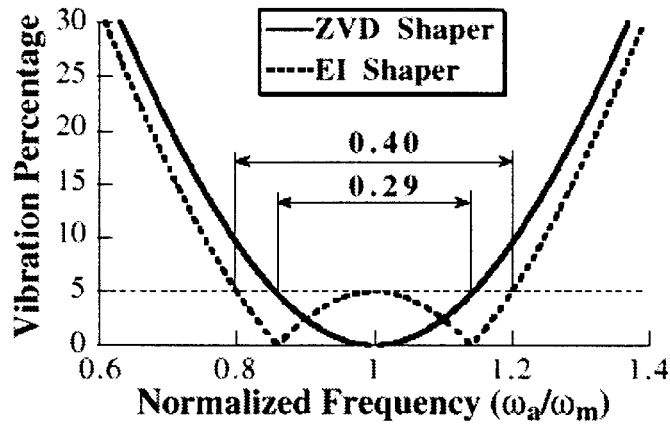
Figure 5: Sensitivity Curves for Common Shapers. [13]

Unfortunately, we cannot always simply select the most robust shaper, because there are other performance specifications of concern. One such characteristic is the shaper's rise time, which in most cases is inversely related to the shaper's insensitivity. See Figure 6. The shapers indicated as points on the curve are those that locally minimize this trade-off.
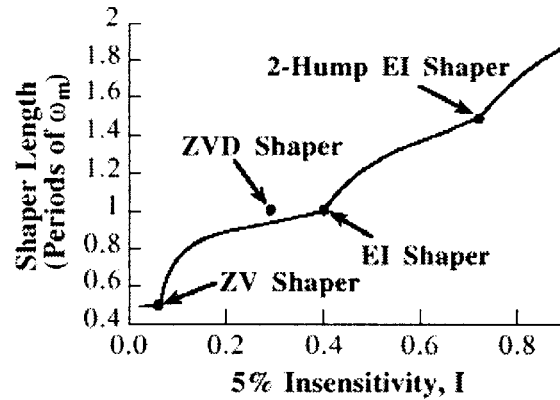
**Figure 6: Trade-off Between Rise Time and
Insensitivity for Different Input Shapers [16].**

In order to improve the rise time, while maintaining the same level of robustness,

shapers that contain negatively-valued impulses can be constructed. These negative

shapers can provide a substantial time savings, but they have the possibility of exciting

unmodeled high modes and causing actuator saturation. One type of negative shaper that

greatly reduces the possibility of actuator saturation is formed by limiting the magnitude

of each impulse to a value of 1. These Unity-Magnitude (UM) shapers can be used with

a great variety of unshaped reference commands without the possibility of actuator

saturation [14]. The job of the input shaper designer is to consider these characteristics

and the trade-offs corresponding to the desired performance.

3. Shaper Selection

3.1 What the Expert System Knows

The expert system contains performance information for six different input

shapers: ZV [17-19], EI [13, 17], Two-Hump EI (EI2HUMP) [15], and the Negative

Unity Magnitude [14] versions of the above shapers (NEGUMZV, NEGUMEI, and

NEGUM2EI). The information about each shaper can be divided into two categories: 1)

information obtained by straightforward mathematical calculations, and 2) information

that has been converted from arduous calculations into simple rules (e.g. how each shaper

affects unmodeled high frequencies). To enable the system to use both types of

information, the expert system rules are weighted.

The first type of information described includes the rise time and insensitivity of

each shaper, which are inversely proportional. The table below shows the best to worst

order of these values for the set of shapers. A lower number indicates a better value.

Shaper	Rise Time	Insensitivity
NEGUMZV	1	6
ZV	2	5
NEGUMEI	3	4
EI	4	3
NEGUM2EI	5	2
EI2HUMP	6	1

Table 1: Trade-off Between Rise Time and Insensitivity for the Shapers in the Expert System

The second type of information used in the expert system is specific to the negative

shapers. The expert system knows not to consider negative shapers if there is an

unmodeled high frequency.

12

3.2 Implementation of Shaper Selection

The expert system first queries the user to obtain the characteristics of the flexible system and the desired performance. Initially, every type of shaper is a possible output of the expert system. Each rule examines a specific performance requirement and compares it with how well each shaper meets the requirement, if at all. If a shaper does not satisfy a criterion (e.g. if the rise time is too high), it is removed from the set of possible shapers. However, the system keeps track of the distance from the shaper's performance value to the desired value in the form of a weight valued 0 to 1.

If a shaper is removed because there is an unmodeled high frequency, a weight of 0.8 is added. This ensures that it is not selected unless all of the other shapers are very far from meeting the performance specifications. All other weights are calculated using the following formula:

$$weight = \frac{|s-d|}{|s-d|+a},$$

where s = shaper's value, d = desired value, and a =calibration constant.

This formula maps the distance between shaper value and desired value to a number between 0 and 1. The shape of the mapping curve is determined by the value of a.

The choice of a is somewhat arbitrary, and almost any choice will yield reasonable results. One choice for a that has worked well in practice is made based on the relative values of each shaper characteristic. If we choose a typical value of the difference in rise time (or insensitivity) and then find a comparable difference in insensitivity (or rise time), then we can adjust a so the weights given these particular values are the same. In the expert system, for example, the rise time difference chosen is 1 second. A comparable difference in insensitivity is about 0.2. So, if the value of a is

100 in the rise time equation, then the value of a will be 20 in the insensitivity equation. The example below demonstrates how these values affect the weights:

$$a_{risetime} = 100 \Rightarrow weight_{risetime} = \frac{1}{1+100} = \frac{1}{101}$$

$$a_{insensitivity} = 20 \Rightarrow weight_{insensitivity} = \frac{0.2}{0.2+20} = \frac{1}{101}$$

Of course, since the weight equation is non-linear, no choice of a will be perfect. However, this model provides a simple way to compare shapers, and it has worked well in practice.

After all the rules have been applied, the system examines the set of shapers that have satisfied all of the performance criteria. If this set contains one or more shapers, then it simply chooses the most robust shaper. If the set is empty, then selection is based on the weights. Each time the system removes a shaper from the set of possibilities, the shaper's weight (initially 0) is adjusted. This adjustment is made using the standard method for combining independent probabilities:

$$new\_weight = old\_weight + (1 - old\_weight) \times (added\_weight)$$

This formula modifies the weight while keeping it between 0 and 1. Since a higher weight indicates less desirable shaper characteristics, the shaper with the lowest weight (the closest match) is selected.

14

3.3 Examples of Shaper Selection

This section presents example cases that demonstrate shaper selection within the expert system. In the following example, the user is prompted for system and performance information. The expert system then removes unacceptable input shapers and chooses the most robust shaper from the remaining set.

Input:

```
Lower bound of frequency (Hz): 0.9
Upper bound of frequency (Hz): 1.1
Damping ratio: 0.1
Tolerable vibration (e.g. 0.05 for 5 percent): 0.05
Is there a maximum rise time? (yes/no) no
Are there any unmodeled high frequencies? (yes/no) no
```

Output:

```
Removing ZV- insensitivity is too low.
Removing NEGUMZV- insensitivity is too low.

At least one shaper meets all criteria-- choosing the most robust
one.
The best shaper is EI2HUMP.

Rise time = 1.488926e+00
Insensitivity = 8.610000e-01
(Desired insensitivity was 2.000000e-01, better by 3.305000e+02
percent)

Shaper Values:
time = 0, amplitude = 2.586675e-01
time = 5.159056e-01, amplitude = 3.601195e-01
time = 1.008206e+00, amplitude = 2.729031e-01
time = 1.488926e+00, amplitude = 1.083099e-01
```

Table 2: Example of Shaper Selection Based on Robustness.

The following is an example of the system choosing a shaper based on weights.

Input:

```
Lower bound of frequency (Hz): 0.8
Upper bound of frequency (Hz): 1.2
Damping ratio 1: 0.1
Tolerable vibration (e.g. 0.05 for 5 percent): 0.05
Is there a maximum rise time? (yes/no) yes
What is the maximum rise time? 0.7
Are there any unmodeled high frequencies? (yes/no) no
```

Output:

```
Removing ZV- insensitivity is too low.
Removing EI- rise time is too high.
Removing EI2HUMP- rise time is too high.
Removing NEGUM2EI- rise time is too high.
Removing NEGUM2EI- may exceed actuator limits.
Removing NEGUMEI- rise time is too high.
Removing NEGUMEI- may exceed actuator limits.
Removing NEGUMZV- insensitivity is too low.
Removing NEGUMZV- may exceed actuator limits.

No shaper fully meet all criteria-- choosing the closest one.
ZV weight = 1.586911e-02
EI weight = 2.991027e-03
EI2HUMP weight = 7.827510e-03
NEGUM2EI weight = 7.014673e-01
NEGUMEI weight = 7.001268e-01
NEGUMZV weight = 7.049640e-01
The best shaper is EI.

Rise time = 1
(Desired time was 7.000000e-01, worse by 4.290000e+01 percent)
Insensitivity = 4.765000e-01
(Desired insensitivity was 4.000000e-01, better by 1.910000e+01
percent)

Shaper Values:
time = 0, amplitude = 3.548861e-01
time = 5.046966e-01, amplitude = 4.529947e-01
time = 1, amplitude = 1.921192e-01
```

Table 3: Example of Shaper Selection Based on Weights.

4. Shaper Refinement

4.1 Why Shaper Refinement Is Necessary

A component of the control engineer's expertise lies in redesigning an input shaper after it has been implemented on a machine. The expert sometimes goes through a few rounds of refinement to further improve performance. An input shaper may be refined for several reasons. Sometimes there is a significant modeling error. Testing the shaper on the machine exposes these modeling errors, and the expert can adjust the shaper accordingly. One such error is an unmodeled high frequency. The expert often designs a shaper to eliminate the main source of vibration, generally the lower frequency, only to find that a higher frequency becomes evident because the low mode has been eliminated. The expert can then redesign the shaper to account for both frequencies.

Actuator saturation is a problem that shows up in the form of performance degradation for long moves. That is, the shaping works well when the structure is moved a short distance but not a long distance. The reason is that the longer moves lead to a larger error, and certain types of closed-loop controllers may issue a larger command. This larger command can saturate the actuators. In this case, the expert should avoid shapers that require high actuator output.

Another source of error is an incorrect modeling parameter (frequency or damping), in which case the input shaper might fail to remove enough of the vibration. The expert handles this problem in one of three ways: 1) recalculate the FFT of the residual vibration, 2) lower the guess of the frequency or damping, 3) raise the guess of the frequency or damping. The first option is theoretically the best; however, a lack of

equipment and appropriate data makes it difficult to do on occasion. Options 2 and 3 are not as theoretically pleasing, but they generally work well in the field.

It is sometimes the case that an input shaper works well in one environment, but not in another. For example, changing the configuration of a robotic arm will change its dynamic behavior. In these cases, the expert must redesign the shaper to be more robust.

Finally, a reason for shaper refinement is simply greed. We see that the performance has improved with input shaping, and we want it to be better. Here, the expert can simply go through the same selection process again with more demanding performance specifications.

4.2 Shaper Refinement Using Different Information

After the expert system selects a shaper, the new command is applied to the flexible structure. The performance of the new command may be measured using a number of characteristics in the response including the overshoot, rise time, settling time and others. The way the expert refines the shaper depends on which of these characteristics may be measured. If, for example, the data for the complete response is available, we can directly measure the frequency and damping.

In many cases, however, only certain pieces of information about the response are available. One important piece of information that is often available is the overshoot. If we know the exact value of the overshoot, we can calculate how far off the modeling frequency is from the actual frequency. If, on the other hand, we only know that the overshoot is too large or small enough, we can make blind adjustments to the frequency and damping modeled in the shaper and re-test it to see if the response has improved.

Sometimes no frequency and damping adjustments yield an adequate response. A number of modeling errors can be the cause. For example, there may be an unmodeled high frequency or the actuators may be near saturation. The shaper selection portion of the expert system is designed to avoid using shapers that exacerbate these problems.

4.3 Refinement Through Blind Search

As mentioned, if we have no information about the response (other than that it is not good enough), we can make reasonable adjustments to our model and re-test the appropriately adjusted shaper. The expert system does this using a hill-climbing algorithm [23] where the frequency and damping are the fitness landscape parameters and the overshoot is the metric used to guide the search. If the overshoot cannot be measured, then the expert system relies on the user to determine if the response has gotten better or worse.

A hill-climbing algorithm is well suited to this task, because the shape of the fitness landscape is smooth and monotonically decreasing towards the actual system parameters. As Figure 7 illustrates, the overshoot of the response decreases as the frequency and damping ratio become more accurate. Furthermore, the overshoot decreases as each individual parameter becomes more accurate. Therefore, we may search for the value of one parameter and then search for the other without having to backtrack and re-adjust the first.
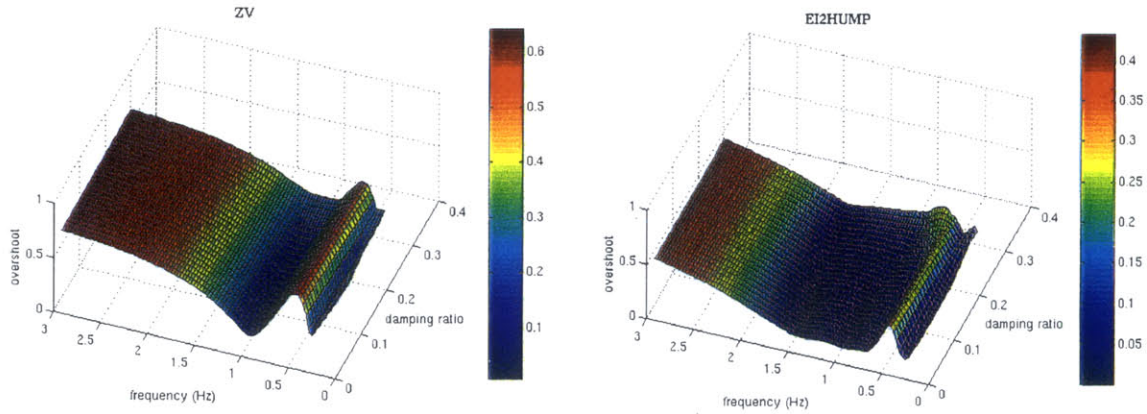
Figure 7: Fitness Landscapes of Modeling Errors for Two Shapers.

At model frequencies that vary greatly from the actual frequency, the shape of the overshoot curve may not decrease towards the correct value. However, in the vast majority of cases, the frequency is known with enough accuracy that this is not an issue.

The EI and NEGUMEI shapers have fitness landscapes in which the overshoot is smaller when the modeling frequency is slightly inaccurate. This is because these shapers are designed to stay below a certain level of vibration for range of frequencies. Figure 5 shows that when the modeling frequency, ω_m, equals the actual frequency, ω_m, the EI shaper produces some small amount of vibration; however, when the actual frequency is slightly greater than or slightly less than the modeling frequency, the vibration goes to zero. The curve for the NEGUMEI shaper is similar.

Because these shapers allow for a low level of vibration, the expert system will also allow the user to specify an allowable overshoot in the response. The percent vibration is essentially a measurement of both the overshoot and settling time in the response. Therefore, a natural choice for the allowable overshoot would be a number that is slightly less than the percent vibration used to calculate the shaper. Using this value,

the algorithm will find a frequency within the range allowed by the shaper. If the
algorithm is pushed to search for a better response, then it will move further and further
towards one of the frequencies at which the overshoot is minimal (see Figure 8).
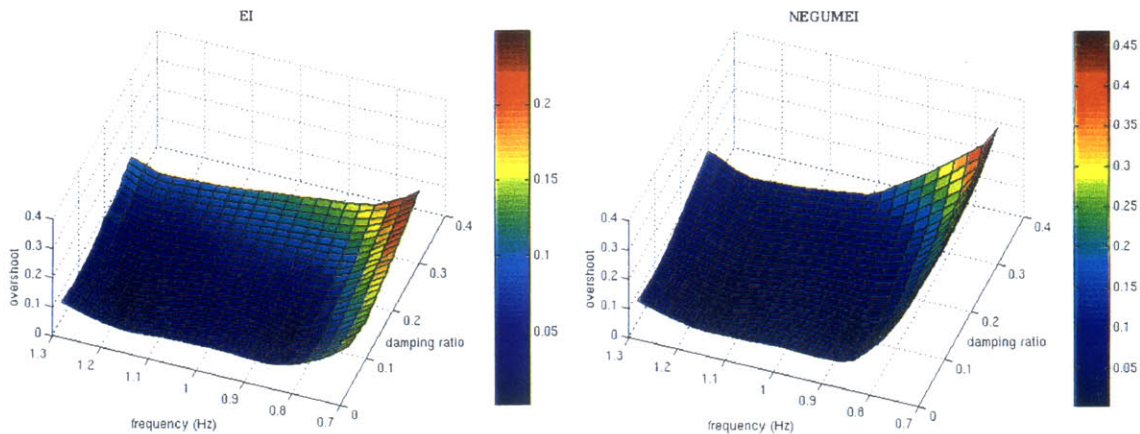
**Figure 8: Fitness Landscapes of Modeling Errors for the EI and NEGUMEI Shapers.
The refinement algorithm moves towards the frequencies in the valleys.**

The refinement algorithm searches for the correct frequency and then searches for ·
the correct damping ratio. First, the algorithm measures the overshoot in the response
based on the initial model. Next, the model frequency is incremented (by 10 percent of
the initial model frequency). The new shaper is calculated with this new frequency, and
it is re-tested. Again, the overshoot of the response is measured. If the overshoot is less
than its previous value, then the search is headed in the right direction, so we continue
with the same adjustment to the modeling frequency. If, however, the overshoot is
greater than its previous value, we need to change the search direction. If we have not
already tested the new direction, we simply re-adjust the frequency and re-test it in the
manner described. If we have already tried that direction (indicating that we are close to
the correct answer), the search increment is divided by two before re-testing. The flow

chart in Figure 9 illustrates this process. The refinement process ends when either the

user chooses to stop, the user-specified overshoot is reached, or the minimum search

increment (equal to 1 percent of the initial model frequency) is reached (indicating that

the frequency value is as close to the actual value as the algorithm allows). The damping

ratio is then determined in the same way.
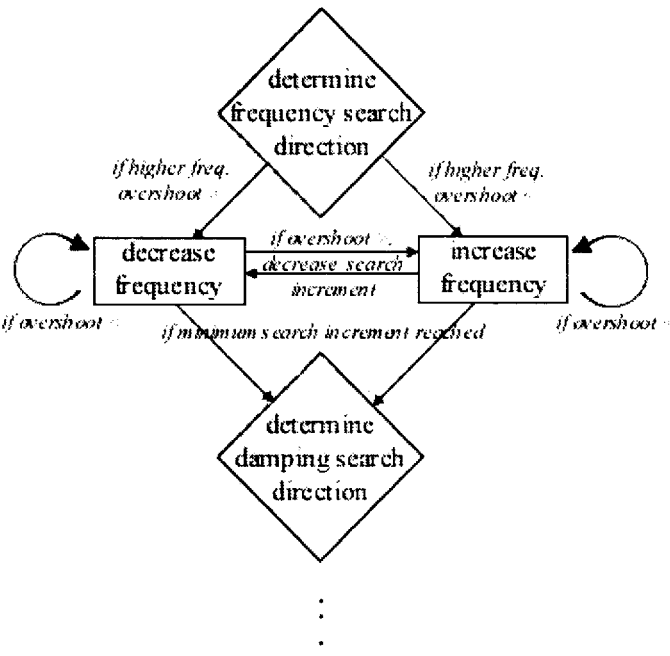
Figure 9: Flow Chart of the Refinement Process.

4.4 Examples of Refinement Through Blind Search

The following examples demonstrate the blind refinement process. The inputs to

the procedure are the shaper name, the model frequency and damping, and the actual

frequency and damping. The numbers displayed by the program are the successive

values for the overshoot in the response. The question "Refine shaper to minimum

overshoot?" asks the user if he would like the program to automatically refine the shaper

to the minimum error. If the user answers "no", then the user may either specify an overshoot value at which the program will automatically stop or continue and tell the program to stop along the way. When the refinement process has finished, the response to the final refined shaper is plotted. The new model values along with the number of rounds it took to determine them are printed.

The following example shows the expert system refining a model with an inaccurate frequency. The algorithm adjusts the model values until the overshoot is minimal. Figure 10 shows the response at each round of refinement. The solid line shows the original response. The dotted lines show the refined responses.

Input:

```
Shaper: ZV
Model frequency: 1
Model damping: 0.1
Actual frequency: 1.34
Actual damping: 0.1
Refine shaper to minimum overshoot? (1=yes, 0=no): 1
```

Output:

```
f = 1, overshoot = 2.712094e-01
f = 1.100000e+00, overshoot = 1.790352e-01
NEED TO INCREASE MODEL FREQUENCY.
f = 1.200000e+00, overshoot = 9.713348e-02
f = 1.300000e+00, overshoot = 2.574945e-02
f = 1.400000e+00, overshoot = 4.916754e-02
f = 1.250000e+00, overshoot = 6.015680e-02
f = 1.325000e+00, overshoot = 9.481576e-03
f = 1.350000e+00, overshoot = 8.506880e-03
f = 1.375000e+00, overshoot = 2.919559e-02
f = 1.337500e+00, overshoot = 1.581832e-03
f = 1.325000e+00, overshoot = 9.481576e-03
Frequency determined to be 1.337500e+00.
Now determining correct damping ratio.
z = 1.100000e-01, overshoot = 1.362726e-02
z = 1.000000e-01, overshoot = 1.581832e-03
z = 9.000000e-02, overshoot = 9.586028e-03
z = 1.050000e-01, overshoot = 6.950562e-03
z = 9.750000e-02, overshoot = 2.682104e-03
z = 1.012500e-01, overshoot = 1.937886e-03
Damping ratio determined to be 1.000000e-01.

Frequency=1.337500e+00, Damping=1.000000e-01
Values determined after 16 rounds.
```

Table 4: Example of Blind Refinement of the ZV Shaper.

Figure 10: Blind Refinement of the ZV Shaper.
The initial frequency was inaccurate.
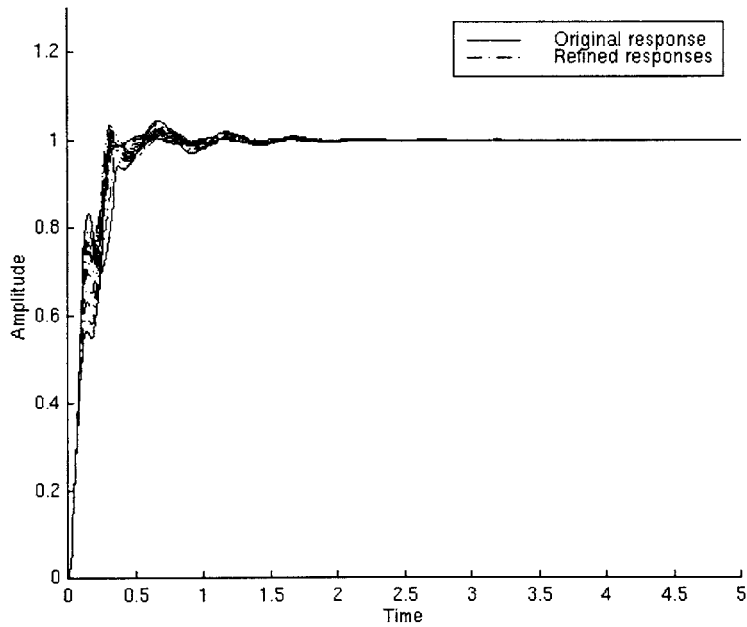
Figure 11: Blind Refinement of the NEGUM2EI Shaper.
The initial damping ratio was inaccurate.

In the next example, the damping is inaccurate. Figure 11 shows the response at each round of refinement.

Input:

```
Shaper: NEGUM2EI
Model frequency: 2
Model damping: 0.2
Actual frequency: 2
Actual damping: 0.13
Refine shaper to minimum overshoot? (1=yes, 0=no): 1
```

Output:

```
f = 2, overshoot = 3.350836e-02
f = 2.200000e+00, overshoot = 4.001649e-02
NEED TO DECREASE MODEL FREQUENCY.
f = 2, overshoot = 3.350836e-02
f = 1.800000e+00, overshoot = 4.173813e-02
f = 2.100000e+00, overshoot = 3.418916e-02
f = 1.950000e+00, overshoot = 3.956825e-02
f = 2.025000e+00, overshoot = 3.109508e-02
f = 2.050000e+00, overshoot = 3.319406e-02
Frequency determined to be 2.025000e+00.
Now determining correct damping ratio.
z = 2.200000e-01, overshoot = 4.448897e-02
z = 2.000000e-01, overshoot = 3.109508e-02
z = 1.800000e-01, overshoot = 2.159849e-02
z = 1.600000e-01, overshoot = 1.326832e-02
z = 1.400000e-01, overshoot = 6.058415e-03
z = 1.200000e-01, overshoot = 5.568910e-03
z = 1.000000e-01, overshoot = 7.560124e-03
z = 1.300000e-01, overshoot = 5.057609e-03
z = 1.400000e-01, overshoot = 6.058415e-03
z = 1.250000e-01, overshoot = 5.466962e-03
z = 1.325000e-01, overshoot = 5.051369e-03
z = 1.350000e-01, overshoot = 5.223154e-03
Damping ratio determined to be 1.325000e-01.

Frequency=2.025000e+00, Damping=1.325000e-01
Values determined after 19 rounds.
```

Table 5: Example of Blind Refinement of the NEGUM2EI Shaper.

In the next example, both the frequency and damping are inaccurate.

Input:

```
Shaper: NEGUMZV
Model frequency: 2
Model damping: 0.1
Actual frequency: 1.5
Actual damping: 0.14
Refine shaper to minimum overshoot? (1=yes, 0=no): 1
```
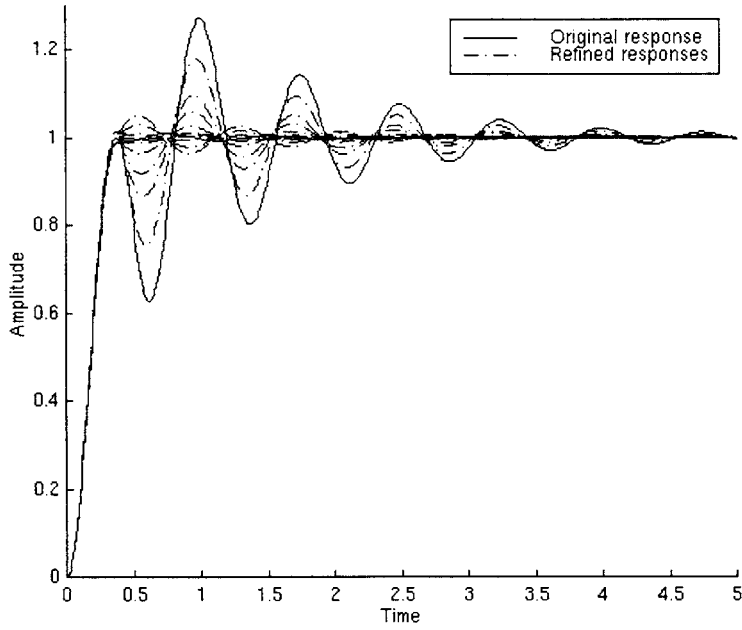
Output:

```
f = 2, overshoot = 2.649433e-01
f = 2.200000e+00, overshoot = 3.269744e-01
NEED TO DECREASE MODEL FREQUENCY.
f = 2, overshoot = 2.649433e-01
f = 1.800000e+00, overshoot = 1.829935e-01
f = 1.600000e+00, overshoot = 7.880479e-02
f = 1.400000e+00, overshoot = 6.854992e-02
f = 1.200000e+00, overshoot = 2.085390e-01
f = 1.500000e+00, overshoot = 3.739382e-02
f = 1.600000e+00, overshoot = 7.880479e-02
f = 1.450000e+00, overshoot = 4.575253e-02
f = 1.525000e+00, overshoot = 4.270990e-02
Frequency determined to be 1.500000e+00.
Now determining correct damping ratio.
z = 1.100000e-01, overshoot = 2.793588e-02
z = 1.200000e-01, overshoot = 1.846463e-02
z = 1.300000e-01, overshoot = 8.977920e-03
z = 1.400000e-01, overshoot = 1.001286e-03
z = 1.500000e-01, overshoot = 1.539920e-02
z = 1.350000e-01, overshoot = 4.228092e-03
z = 1.425000e-01, overshoot = 4.530816e-03
z = 1.387500e-01, overshoot = 7.255472e-04
z = 1.375000e-01, overshoot = 1.865369e-03
Damping ratio determined to be 1.387500e-01.
```

Frequency=1.500000e+00, Damping=1.387500e-01
```
Values determined after 19 rounds.
```

Table 6: Example of Blind Refinement of the NEGUMZV Shaper.

4.5 Refinement Through Guided Search

If we can measure the overshoot in the response, we can calculate the percent

error in the modeling frequency. We can do this by calculating an overshoot curve for

each shaper beforehand. The curve is independent of the frequency (it only depends on

the error in the modeling frequency), so we only have to calculate it once for each shaper.

In Figure 12, we see that given the overshoot we can determine the ratio of the actual

frequency, ω_a, to the modeling frequency, ω_m. For example, if we are using an EI shaper,

and the overshoot is 5 percent, then the actual frequency is either ω_m, $0.8 \times \omega_m$, or $1.2 \times$

ω_m.
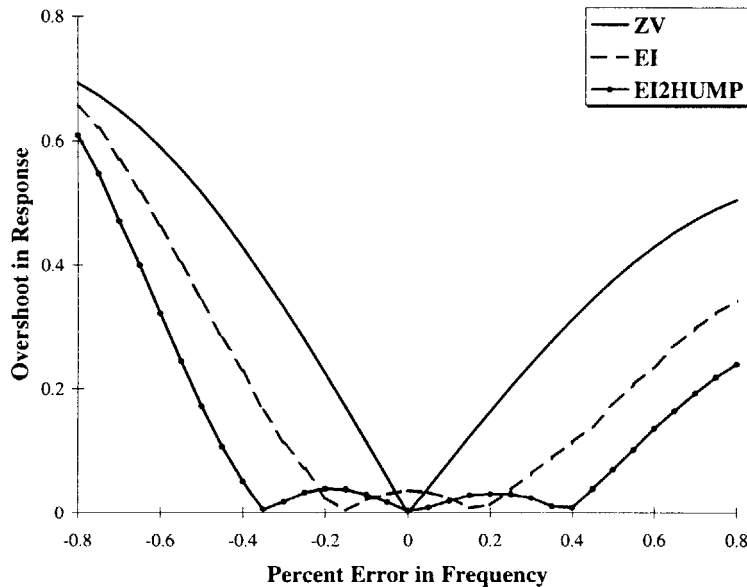
Figure 12: Overshoot Curves For Common Shapers.

The expert system can use this information in the refinement algorithm.

However, since there are at least two values of the frequency error that go with each

value of the overshoot, we need to determine which one is correct. For the EI,

NEGUMEI, EI2HUMP, and NEGUM2EI shapers, we can ignore the values closest to 0

28

percent error, since the overshoot here is already below the allowed amount. So, we are left to choose between two frequency values. We do this with one round of search. We try the 10 percent above the model frequency, and if the performance has improved, then the higher frequency is correct. If not, we go to the lower frequency.

The damping ratio affects the width of the sensitivity curves (see Figure 13). So, if there is also a modeling error in the damping, there will be a small error in the frequency that the algorithm determines. However, even with this error present, the guided refinement algorithm will still be an improvement over the blind algorithm.
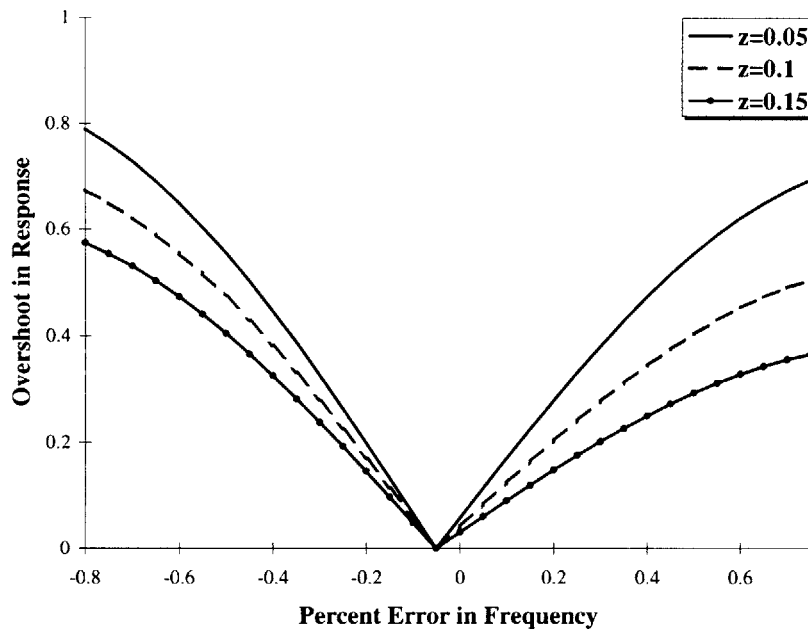
Figure 13: Overshoot Curves for Different Damping Ratios

Once the refinement algorithm jumps to the correct frequency on the sensitivity curve, the blind search is used with a small search increment (equal to 5 percent of the model frequency). It is used to find the frequency more precisely and then to determine the damping.

4.6 Examples of Refinement Through Guided Search

The following examples demonstrate shaper refinement through guided search. The inputs are the same as for the blind search. The output shows that the guided search determines the frequency very accurately after it determines the search direction. We can also see that as the damping error increases, the error in this first frequency determined also increases.

Input:

```
Shaper: ZV
Model frequency: 1.8
Model damping: 0.09
Actual frequency: 1.4
Actual damping: 0.13
Refine shaper to minimum overshoot? (1=yes, 0=no): 1
```

Output:

```
f = 1.800000e+00, overshoot = 2.262298e-01
f = 1.890000e+00, overshoot = 2.617744e-01
NEED TO DECREASE MODEL FREQUENCY.
f = 1.458000e+00, overshoot = 5.281486e-02
f = 1.368000e+00, overshoot = 3.727549e-02
f = 1.278000e+00, overshoot = 7.625300e-02
f = 1.413000e+00, overshoot = 3.589432e-02
f = 1.458000e+00, overshoot = 5.281486e-02
f = 1.390500e+00, overshoot = 3.412975e-02
f = 1.368000e+00, overshoot = 3.727549e-02
Frequency determined to be 1.390500e+00.
Now determining correct damping ratio.
z = 9.900000e-02, overshoot = 2.647179e-02
z = 1.080000e-01, overshoot = 1.893301e-02
z = 1.170000e-01, overshoot = 1.170147e-02
z = 1.260000e-01, overshoot = 5.810682e-03
z = 1.350000e-01, overshoot = 6.416205e-03
z = 1.215000e-01, overshoot = 8.421630e-03
z = 1.282500e-01, overshoot = 5.008616e-03
z = 1.305000e-01, overshoot = 4.707871e-03
z = 1.327500e-01, overshoot = 4.909240e-03
z = 1.293750e-01, overshoot = 4.791946e-03
Damping ratio determined to be 1.305000e-01.

Frequency=1.390500e+00, Damping=1.305000e-01
Values determined after 18 rounds.
```

Table 7: Example of Guided Refinement of the ZV Shaper.

In the following example, the EI2HUMP shaper is refined until the response overshoot is below 0.05. Since the shaper has been designed stay below 0.05 for a wide range of frequencies, the algorithm finds an adequate solution very quickly.

Input:

```
Shaper: EI2HUMP
Model frequency: 1.6
Model damping: 0.09
Actual frequency: 1
Actual damping: 0.01
Refine shaper to minimum overshoot? (1=yes, 0=no): 0
Specify acceptable overshoot? (1=yes, 0=no): 1
Acceptable overshoot: 0.05
```

Output:

```
f = 1.600000e+00, overshoot = 6.919528e-02
f = 1.680000e+00, overshoot = 9.581014e-02
NEED TO DECREASE MODEL FREQUENCY.
f = 9.440000e-01, overshoot = 5.231258e-02
f = 8.640000e-01, overshoot = 8.194353e-02
f = 9.840000e-01, overshoot = 4.567457e-02
Frequency determined to be 9.840000e-01.
Now determining correct damping ratio.
Damping ratio determined to be 9.000000e-02.

Frequency=9.840000e-01, Damping=9.000000e-02
Values determined after 4 rounds.
```

Table 8: Example of Guided Refinement of the EI2HUMP Shaper.

In the next example, the NEGUMEI shaper is refined until the overshoot is below 0.0125. Again, this shaper is designed to be below this value for a range of frequencies. So once the modeling frequency is within that range, the algorithm stops searching.

Input:

```
Shaper: NEGUMEI
Model frequency: 1
Model damping: 0.1
Actual frequency: .7
Actual damping: 0.15
Refine shaper to minimum overshoot? (1=yes, 0=no): 0
Specify acceptable overshoot? (1=yes, 0=no): 1
Acceptable overshoot: 0.0125
```

Output:

```
f = 1, overshoot = 1.336110e-01
f = 1.050000e+00, overshoot = 1.622803e-01
NEED TO DECREASE MODEL FREQUENCY.
f = 7.250000e-01, overshoot = 9.345641e-03
Frequency determined to be 7.250000e-01.
Now determining correct damping ratio.
Damping ratio determined to be 1.000000e-01.

Frequency=7.250000e-01, Damping=1.000000e-01
Values determined after 2 rounds.
```

Table 9: Example of Guided Refinement of the NEGUMEI Shaper.

5. Experiments

In this section, I describe the tests run on the selection and refinement portions of the expert system. I first present the design of the shaper selection tests and their results followed by an explanation of the results. I then present the design and interpretation of the shaper refinement tests.

5.1 Design of Shaper Selection Tests

The shaper selection part of the expert system was tested on a series of inputs, some to test the system's ability to choose the most robust shaper and some to demonstrate the system's ability to choose a shaper based on weights. Clearly, the latter is the more interesting and challenging task.

The inputs were chosen as follows. I first chose one frequency and one damping ratio, so that the values of the rise time and insensitivity for each shaper would remain constant. This allowed me to partition the space of rise times and insensitivities around the shapers. That is, knowing the rise time and insensitivity of each shaper, I could design performance specifications that would include or exclude a certain set of shapers. For a frequency of 1 Hz and a damping ratio of 0.1, the values for each shaper are listed in the table below. The four EI shapers were designed to stay below 5 percent vibration.

Shaper	Rise Time	Insensitivity
NEGUMZV	0.3358	0.0635
ZV	0.5025	0.0775
NEGUMEI	0.7423	0.4085
EI	1.000	0.4765
NEGUM2EI	1.191	0.7510
EI2HUMP	1.489	0.8610

Table 10: Rise Time and Insensitivity Values Used in Shaper Selection Tests.

After selecting a frequency and a damping ratio, I tested the shaper selection

process on seven different rise time values and seven different frequency values, one

value for each of the eight possible sets of shapers. For example, giving no rise time

requirement does not exclude any of the shapers, while specifying a maximum rise time

of 0.8 excludes the EI, NEGUM2EI, and EI2HUMP shapers. Similarly, specifying an

insensitivity of 0.4 excludes the NEGUMZV and ZV shapers. Each of these

combinations was also tested with a "yes" response and a "no" response to the question

about the presence of unmodeled high frequencies.

The results of the tests are summarized in the table below. Some of the names of

the shapers have been abbreviated: EI2HUMP is 2EI, and the negative shapers start with

"N-" instead of "NEGUM". A shaper name in regular font indicates that at least one

shaper met the performance specifications; it was selected because it was the most robust

of the set. A bold name indicates that no shaper met all of the performance

specifications; it was selected because it had the least negative weight. The results

presented show that the expert system makes both types of selections accurately.

Rise time <	Unmodeled high mode?	Insensitivity >						
		0.06	0.07	0.4	0.45	0.7	0.8	0.9
1.5	No	2EI	2EI	2EI	2EI	2EI	2EI	**2EI**
	Yes	2EI	2EI	2EI	2EI	2EI	2EI	**2EI**
1.2	No	N-2EI	N-2EI	N-2EI	N-2EI	N-2EI	**N-2EI**	**N-2EI**
	Yes	EI	EI	EI	EI	**2EI**	**2EI**	**2EI**
1.1	No	EI	EI	EI	EI	**N-2EI**	**N-2EI**	2EI
	Yes	EI	EI	EI	EI	**2EI**	**2EI**	2EI
0.9	No	N-EI	N-EI	N-EI	**EI**	**N-2EI**	**N-2EI**	2EI
	Yes	ZV	ZV	**EI**	**EI**	**2EI**	**2EI**	2EI
0.6	No	ZV	ZV	**N-EI**	**N-EI**	**N-2EI**	**N-2EI**	**N-2EI**
	Yes	ZV	ZV	**EI**	**EI**	**2EI**	**2EI**	2EI
0.4	No	N-ZV	**N-ZV**	**N-EI**	**N-EI**	**N-2EI**	**N-2EI**	2EI
	Yes	**ZV**	**ZV**	**EI**	**EI**	**2EI**	**2EI**	2EI
0.3	No	**N-ZV**	**N-ZV**	**N-EI**	**N-2EI**	**N-2EI**	**N-2EI**	2EI
	Yes	**ZV**	**ZV**	**EI**	**EI**	**2-EI**	**2EI**	2EI

Table 11: Results of Shaper Selection Tests.
Regular font indicates that the selection was made based on robustness.
Bold font indicates that the selection was made based on weights.

5.2 Interpretation of Shaper Selection Tests

5.2.1 Selection based on robustness

As indicated in the table of results, when the rise time and insensitivity specifications do not exclude all of the shapers, the expert system chooses the most robust shaper of those remaining. For example, when the rise time must be below 1.5 and the insensitivity must be at least 0.06, then any of the shapers may be used. Because

the EI2HUMP shaper has the highest insensitivity, it is selected. As another example, if

the rise time must be less than 1.1 and the insensitivity must be at least 0.4, then the EI

shaper selected. It is the most robust of the shapers that meet both. The tables below

illustrate the selection process for these examples:

Specification	Rise time < 1.5		Insensitivity > 0.06		
Shapers Remaining (most to least robust)	{2EI, N-2EI, EI, N-EI, ZV, N-ZV}	$\rightarrow$	{2EI, N-2EI, EI, N-EI, ZV, N-ZV}	$\rightarrow$	**Final Selection:** 2EI

Specification	Rise time < 1.1		Insensitivity > 0.06		
Shapers Remaining (most to least robust)	{EI, N-EI, ZV, N-ZV}	$\rightarrow$	{EI, N-EI}	$\rightarrow$	**Final Selection:** EI

Table 12: Shaper Selection Based on Robustness.

5.2.2 Selection based on weights

If the rise time and insensitivity specifications exclude all of the shapers, then the

shaper that most closely meets these specifications is chosen. The weights described in

section 3.2 are used to determine which shaper is the closest. For example, if the rise

time must be less than 1.1, then we exclude the NEGUM2EI and EI2HUMP shapers.

Then, if the insensitivity must be greater than 0.7, then we exclude the remaining shapers.

The expert system then tries to find the shapers that most closely meets both

specifications. We see that the EI shaper meets the rise time specification, but its

insensitivity is far from the desired value. The NEGUM2EI meets the insensitivity

specification and is close to the rise time specification. The EI2HUMP shaper also has

the desired insensitivity, but the rise time is much longer than the desired value. The NEGUM2EI shaper is selected, because the weighting system determines that it is the closest match. It is clearly better than the EI2HUMP shaper, since it more closely meets the rise time specification. It is better than the EI shaper, because a difference of 0.09 seconds in rise time is much less significant that a difference of 0.2235 in insensitivity.

5.2.3 Presence of unmodeled high frequencies

When an unmodeled high mode is present, the expert system excludes the three negative shapers. So, if the selection without the unmodeled mode is a negative shaper, it changes to a positive shaper with the unmodeled mode. In some cases, the positive shaper also meets the performance criteria, but in other cases, the selection is made based on weights. For example, if the rise time must be less than 0.9 and the insensitivity must be greater than 0.07, then the selection is ZV when an unmodeled mode is present. The ZV shaper is the only shaper that meets all of the criteria. On the other hand, if the insensitivity must be greater than 0.4, then no positive shaper is a possibility. The expert system chooses the EI shaper based on weights. It is chosen over the EI2HUMP shaper, because its rise time is faster. It is chosen over the ZV shaper, because the EI's 0.1 difference in rise time is not as significant as the ZV's 0.3225 difference in insensitivity.

Specification	Rise time < 0.9	Insensitivity > 0.07		Unmodeled high mode		
Shapers Remaining (most to least robust)	{N-EI, ZV, N-ZV}	{ZV, N-EI}	→	{ZV}	→	Final Selection: ZV

Specification	Rise time < 0.9	Insensitivity > 0.4		Unmodeled high mode		
Shapers Remaining (most to least robust)	{N-EI, ZV, N-ZV}	{N-EI}	→	{ }	→	Final Selection: EI

Table 13: Shaper Selection with Unmodeled High Frequencies.

5.3 Design of Shaper Refinement Tests

Both the blind refinement and guided refinement processes were able to find an adequate model for a wide range of inputs. So, testing this part of the expert system consisted mainly of comparing the number of trials taken by the blind and guided searches. In almost every case, the guided search was an improvement over the blind search. However, the amount of improvement depended on the values of the system variables—the shaper used, the maximum overshoot specified, the frequency error, and the damping error.

Tests were run on a many combinations of system variables. For each shaper, frequency errors of −50 percent to +50 percent in 10 percent increments were tested. Also, 0, −50 percent and +50 percent damping errors were tested. The values used for the actual frequency and actual damping ratio were 1 Hz and 0.1, respectively. The maximum overshoot values tested were 0.02, 0.03, 0.04, and 0.05 for the ZV and NEGUMZV shapers, while values of 0.0125 and 0.05 were used for the EI, NEGUMEI,

EI2HUMP, and NEGUM2EI shapers. The vibration limits for the latter four shapers were set equal to the specified overshoot.

A maximum overshoot was specified for all of the tests, because this more accurately reflects how refinement is done in the field. More refinement trials are spent trying to achieve a desired performance rather than simply searching down to the algorithm's minimum search increment, which can be changed arbitrarily. The test results will now more accurately reflect the difference between the two search methods. The results are summarized in the tables below.

The set of tables below shows the average number of rounds for the blind and guided searched for each shaper. Tables are presented for each damping error. The data was collected over all of the frequency errors and maximum overshoot values tested.

model damping = actual damping

Shaper	Average Blind Rounds	Average Guided Rounds
NEGUMZV	3.50	2.00
ZV	3.50	2.00
NEGUMEI	2.30	1.40
EI	2.20	1.40
NEGUM2EI	1.35	1.10
EI2HUMP	0.60	0.60

model damping = 0.5 × actual damping

Shaper	Average Blind Rounds	Average Guided Rounds
NEGUMZV	12.77	14.52
ZV	16.91	19.05
NEGUMEI	7.18	6.23
EI	6.00	5.91
NEGUM2EI	2.09	1.86
EI2HUMP	1.36	1.73

model damping = 1.5 × actual damping

Shaper	Average Blind Rounds	Average Guided Rounds
NEGUMZV	8.02	7.61
ZV	6.70	6.64
NEGUMEI	2.27	1.59
EI	2.00	1.27
NEGUM2EI	1.23	1.09
EI2HUMP	0.27	0.18

Table 14: Average Number of Rounds for Blind and Guided Refinement.

The next set of tables shows the average number of rounds for the blind and guided searches for different values of the frequency error. The data is averaged across the different overshoot values. Only the ZV shaper data is presented, because the other shapers followed the same pattern.

model damping = actual damping

Frequency Error	Average Blind Rounds	Average Guided Rounds
-0.5	6	2
-0.4	5	2
-0.3	4	2
-0.2	3	2
-0.1	2	2
0	0	0
0.1	1	2
0.2	2	2
0.3	3	2
0.4	4	2
0.5	5	2

model damping = 0.5 × actual damping

Frequency Error	Average Blind Rounds	Average Guided Rounds
-0.5	15	20.25
-0.4	14	20
-0.3	16.5	16.5
-0.2	14.25	17.25
-0.1	13	17.5
0	13	15
0.1	18.25	14
0.2	17.5	18
0.3	22.25	20
0.4	23	24
0.5	22.5	27

model damping = 1.5 × actual damping

Frequency Error	Average Blind Rounds	Average Guided Rounds
-0.5	9.25	7.75
-0.4	8.25	8.5
-0.3	7.25	5.75
-0.2	6.25	6.75
-0.1	5.25	6.5
0	3.75	4.75
0.1	4.75	5.25
0.2	5.75	6.5
0.3	6.75	4.75
0.4	7.75	6.5
0.5	8.75	8.5

Table 15: Average Number of Rounds for Different Frequency Errors.

5.4 Interpretation of Shaper Refinement Tests

5.4.1 Different frequency and damping errors

As expected, the refinement process took fewer rounds when the frequency and damping were more accurate.

5.4.2 Blind and guided refinement

As expected, the guided search was an improvement over the blind search in most cases. As the results indicate, when the damping was known exactly, the frequency was found by the guided search immediately after determining the search direction (two rounds total). When there was a damping error, both search methods took more rounds. Furthermore, as the damping error increased, the guided search became worse in comparison to the blind search.

5.4.3 Different shapers

As Table 14 indicates, as the shaper's insensitivity increases, the number of rounds for each search method decreases. Furthermore, the percent improvement in the guided search decreases. For example, when using the NEGUM2EI shaper, we are generally already very close to an acceptable frequency, so a blind search will take very few rounds. Therefore, a guided search is much more useful for the ZV and NEGUMZV shapers than for those with a high insensitivity.

6. Conclusions

An expert system has been developed to select and refine input shapers. Given the model of a mechanical system and a set of performance requirements, the expert system can select the best shaper from a variety of shapers, based on rules capturing their known trade-offs. Once the shaper has been selected and applied, if the response is fed back into the expert system, the modeling parameters can be refined and used to design a new input shaper. The result is an improved system model and enhanced performance.

The shaper selection process captures the trade-offs between the different shaper characteristics and performance criteria very well. Furthermore, it is flexible to changes. If an output does not match the expert's selection, the weight equation used to determine it can be adjusted. Although the weight equation used can never be perfect, it provides a simple and adequate model that works well in practice.

The shaper refinement part of the expert system also works very well. The blind search is capable of refining the shaper until the response satisfies the user. The process took fewer rounds when the frequency and damping were more accurate, as expected. When a particular maximum overshoot was specified, the algorithm took far fewer rounds for shapers with a higher insensitivity. This was also expected, because with a higher insensitivity, a wide range of frequencies yields low levels of vibration.

The guided refinement algorithm used the value of the overshoot to determine the actual frequency. This proved to be an improvement over the blind search, in that it took fewer rounds to determine the actual frequency. The guided search also worked better when the damping ratio was known. This was expected, because the algorithm uses the guess for the damping ratio when determining the actual frequency.

The overall improvement of the guided search over the blind search was clearer when a less strict maximum overshoot was specified. Doing this, we remove the rounds of fine-tuning within each search and keep only those which make large performance improvements. The guided search is better in this respect because the first frequency adjustment it makes is often very accurate.

This improvement was even more obvious when using shapers with low insensitivity, like the positive and negative ZV shapers. For these, the blind search would take many more rounds to locate the small range of adequate frequencies than the guided search. The improvement was not significant for the shapers with very high insensitivities. For the 2-hump EI (EI2HUMP) and the negative 2-hump (NEGUM2EI), it did not take very many rounds of blind search to located the wide range of adequate frequencies. The guided search generally took only one or two fewer trials.

7. References

[1] J.T. Feddema, "Digital Filter Control of Remotely Operated Flexible Robotic Structures," *American Control Conf.* San Francisco, CA, 1993, pp. 2710-2715.

[2] J.M. Hyde and W.P. Seering, "Inhibiting Multiple Mode Vibration in Controlled Flexible Systems," *American Control Conf.* Boston, MA, 1991.

[3] S.D. Jones and A.G. Ulsoy, "Control Input Shaping for Coordinate Measuring Machines," *American Control Conf.* Baltimore, MD, 1994, pp. 2899-2903.

[4] F. Khorrami, S. Jain, and A. Tzes, "Adaptive Nonlinear Control and Input Preshaping for Flexible-Link Manipulators," *American Control Conf.* San Francisco, CA, 1993, pp. 2705-2709.

[5] D.P. Magee, "Optimal Arbitrary Time-Delay Filtering to Minimize Vibration in Elastic Manipulator Systems," Ph.D., Georgia Institute of Technology, 1996.

[6] D.P. Magee and W.J. Book, "Filtering Micro-Manipulator Wrist Commands to Prevent Flexible Base Motion," *American Control Conf.* Seattle, WA, 1995, pp. 924-928.

[7] B.R. Murphy and I. Watanabe, "Digital Shaping Filters for Reducing Machine Vibration," *IEEE Transactions on Robotics and Automation*, April, 1992, pp. 285-289.

[8] M.W. Noakes and J.F. Jansen, "Generalized Inputs for Damped-Vibration Control of Suspended Payloads," *Robotics and Autonomous Systems*, 2, 1992, pp. 199-205.

[9] L.Y. Pao, T.N. Chang, and E. Hou, "Input Shaper Designs for Minimizing the Expected Level of Residual Vibration in Flexible Structures," *American Control Conf.* Albuquerque, NM, 1997, pp. 3542-3546.

[10] N.C. Singer and W.P. Seering, "Preshaping Command Inputs to Reduce System Vibration," *J. of Dynamic Sys., Measurement, and Control*, March, 1990, pp. 76-82.

[11] T. Singh and G.R. Heppler, "Shaped Input Control of a System With Multiple Modes," *ASME Journal of Dynamic Systems, Measurement, and Control*, September, 1993, pp. 341-437.

[12] W.Singhose, *Command Generation for Flexible Systems.* Ph.D. Thesis, Department of Mechanical Engineering, Massachusetts Institute of Technology, May 1997, p. 22.

[13] W. Singhose, W. Seering, and N. Singer, "Residual Vibration Reduction Using Vector Diagrams to Generate Shaped Inputs," *J. of Mechanical Design*, June, 1994, pp. 654-659.

[14] W. Singhose, N. Singer, and W. Seering, "Time-Optimal Negative Input Shapers," *J. of Dynamic Systems, Measurement, and Control*, June, 1997, pp. 198-205.

[15] W.E. Singhose, *et al.*, "Vibration Reduction Using Multi-Hump Input Shapers," *J. of Dynamic Systems, Measurement, and Control*, June, 1997, pp. 320-326.

[16] W.E. Singhose, W.P. Seering, and N.C. Singer, "Input Shaping for Vibration Reduction with Specified Insensitivity to Modeling Errors," *Japan-USA Sym. on Flexible Automation*. Boston, MA, 1996.

[17] W.E. Singhose, W.P. Seering, and N.C. Singer, "Shaping Inputs to Reduce Vibration: A Vector Diagram Approach," *IEEE Int. Conf. on Robotics and Automation*. Cincinnati, OH, 1990, pp. 922-927.

[18] O.J.M. Smith, *Feedback Control Systems*. New York: McGraw-Hill Book Co., Inc., 1958, pp. 331-345.

[19] O.J.M. Smith, "Posicast Control of Damped Oscillatory Systems," *Proceedings of the IRE*, September, 1957, pp. 1249-1255.

[20] T.D. Tuttle and W.P. Seering, "A Zero-placement Technique for Designing Shaped Inputs to Suppress Multiple-mode Vibration," *American Control Conf.* Baltimore, MD, 1994, pp. 2533-2537.

[21] A. Tzes and S. Yurkovich, "An Adaptive Input Shaping Control Scheme for Vibration Suppression in Slewing Flexible Structures," *IEEE Transactions on Control Systems Technology*, June, 1993, pp. 114-121.

[22] J. Watkins and S. Yurkovich, "Vibration Control for Slewing Flexible Structures," *American Control Conference*. 1992, pp. 2525-9.

[23] P.H. Winston, *Artificial Intelligence* third edition. Addison-Wesley, Massachusetts. 1992, pp. 70-73.