This is to certify that the

dissertation entitled

SHOCK RESPONSE SPECTRUM AND FATIGUE DAMAGE:
A NEW APPROACH TO PRODUCT FRAGILITY TESTING

presented by

MATTHEW PAUL DAUM

has been accepted towards fulfillment
of the requirements for

PH.D. degree in PACKAGING

Gary Burgess
Major professor

Date 1/29/99

SHOCK RESPONSE SPECTRUM AND FATIGUE DAMAGE: A NEW APPROACH TO PRODUCT FRAGILITY TESTING

By

Matthew Paul Daum

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

School of Packaging

1999

ABSTRACT

SHOCK RESPONSE SPECTRUM AND FATIGUE DAMAGE:
A NEW APPROACH TO PRODUCT FRAGILITY TESTING

By

Matthew Paul Daum

Product fragility assessment in packaging has for years been based on modeling fragile components as linear, undamped spring/mass systems inside a rigid frame. This led to the development of the Damage Boundary Curve (DBC) which evaluates the velocity change and deceleration of an input shock for its damage potential to a product. The DBC has some limitations because of assumptions used in its derivation, including its reliance on square shaped input shocks, requirement of multiple test specimens, and the use of a large, often expensive, shock test equipment. The Shock Response Spectrum (SRS) was borrowed from other engineering disciplines and applied to fragility issues in packaging in part to simplify data gathering for DBCs. SRS uses the response of a component to an input shock rather than the input shock itself, and can be used to eliminate the need for testing whole products, extracting the same fragility information as traditional DBC testing without a shock table. However, a serious limitation in traditional DBC and SRS-generated DBC curves is the assumption that failure of the component occurs in a brittle mode, meaning that all shocks before failure have no effect on the product. Most products are in fact not brittle in nature, but ductile, and so failure may be a cumulative effect. This results in a DBC which must incorporate not only velocity change and deceleration, but number of cycles to failure as well.

The purpose of this study was to develop a mathematical model expanding the SRS technique to account for the ductile nature of many products, using an elastic, perfectly plastic model as an idealization of the stress-strain curve. Components exhibiting brittle behavior can also be handled with this model since the brittle stress-strain curve can be considered to be a special case of the elastic, perfectly plastic curve. Two new testing procedures to obtain product fragility information are outlined and demonstrated on four products, including one method that does not require a shock table. Software was also developed to demonstrate the new SRS-fatigue algorithm and to prove its applicability.

Results show good agreement using the predictions from the new SRS-fatigue model and the new test procedures for finding product fragility information. The result may be considered a new approach to fragility testing which includes not only velocity change and deceleration information, but number of cycles to failure as well. By finding the characteristic material properties of the component, predictions about its response to any kind of input shock can be made.

Dedicated to my loving and supportive wife, my best friend, April.

To my Lord and Savior Jesus Christ, in whom rests all knowledge and truth, and who gives purpose and meaning to life.

ACKNOWLEDGMENTS

Thank you, Dr. Burgess, for helping an ordinary student reach an extraordinary goal.

Thank-you also to my committee members, Dr. S. Paul Singh, Dr. Brian Feeny, and Dr. Robb Clarke.

TABLE OF CONTENTS

LIST OF TABLES x

LIST OF FIGURES xi

CHAPTER 1 1

INTRODUCTION AND LITERATURE REVIEW 1

CHAPTER 2 19

THEORETICAL DEVELOPMENT 19

2.1 FINDING MATERIAL PROPERTIES x_0, x_1 AND F_0 19

2.1.1 Plastic Programmer Method For Determining Material Properties 21

2.1.2 Freefall Method For Determining Material Properties 27

2.1.3 Sources Of Experimental Error 29

2.2 SRS AND DUCTILE MODEL 30

2.3 NUMERICAL SOLUTION 35

2.4 GENERATING FATIGUE DBC CURVES 36

2.5 TEST TO VERIFY PROGRAM ALGORITHM 36

CHAPTER 3 39

MATERIALS AND TEST METHODS 39

3.1 MATERIALS 39

3.2 METHODS 40

3.2.1 Determination Of Natural Frequency 40

3.2.2 Definition Of Failure 46

3.2.3 Test Method For Finding Material Properties From Plastic Programmer Testing 46

3.2.4 Validation Of Results From Plastic Programmer Testing 47

3.2.5 Test Method For Finding Material Properties From Freefall Drops 50

3.2.6 Validation Of Results From Freefall Drops 50

3.2.7 Method For Generating Fatigue DBCs 50

CHAPTER 4 52

RESULTS 52

4.1 RESULTS FOR MATERIAL PROPERTIES FROM PLASTIC PROGRAMMER TESTING 52

4.2 RESULTS FOR VALIDATING MATERIAL PROPERTIES FOUND FROM PLASTIC PROGRAMMER TESTING 54

4.3 RESULTS FOR MATERIAL PROPERTIES FOUND IN FREEFALL DROPS 57

4.4 RESULTS FOR VALIDATING MATERIAL PROPERTIES FOUND FROM FREEFALL DROP TESTING 59

4.5 RESULTS FOR GENERATING FATIGUE DAMAGE BOUNDARY CURVES 59

4.6 DISCUSSION OF ERRORS AND IMPACT ON RESULTS 65

CHAPTER 5 69

CONCLUSIONS AND FUTURE WORK 69

5.1 CONCLUSIONS 69

5.2 FUTURE WORK 71

APPENDICES

APPENDIX A RECOVERING OTHER MODELS 74

APPENDIX B MATERIAL PROPERTIES FROM FREEFALL TESTING ... 77

APPENDIX C SRS PROGRAM CODE 79

REFERENCES 99

LIST OF TABLES

Table 1. Variation in Observed Number of Drops to Failure for Test Models.....26

Table 2a. Comparison of Values, Trapezoid Pulse.....38

Table 2b. Comparison of Values, Half-Sine Pulse.....38

Table 3. Natural Frequency of Test Models.....46

Table 4. Failure Modes and Determination Method.....46

Table 5. Freefall Drop Height Test Configurations.....48

Table 6. Summary of Material Properties From Plastic Programmer Testing.....53

Table 7. Actual and Predicted Drops to Failure, Freefall Testing, PMMA Beams.....56

Table 8. Actual and Predicted Drops to Failure, Freefall Testing, Sheet Metal.....56

Table 9. Summary of Calculated Material Properties From Freefall Testing.....58

Table 10. Comparison of Calculated ΔVcr and Gcr.....59

Table 11. Data For Constructing Fatigue DBCs.....60

Table 12. Variation of x_0 and x_1 From Static Deflection Test.....66

Table 13. Values For Number of Drops to Failure.....66

Table 14. Effect of N on Material Properties x_0 and x_1.....67

LIST OF FIGURES

Figure 1. Product With Fragile Component, Modeled As Spring/Mass System 2

Figure 2. Traditional Damage Boundary Curve 2

Figure 3. SRS Plot From Commercial Software By Lansmont Corporation 6

Figure 4. Shock Response Plot From Commercial Software By Lansmont Corporation 6

Figure 5. Dynamic Force Deflection Curve, Including Damping 9

Figure 6. Material Parameters Describing Ductile Failure 13

Figure 7. Multiple Stretching and Releasing of a Ductile Component 13

Figure 8. Generalized DBCs Incorporating Fatigue Damage 16

Figure 9. Simple Spring/Mass/Dashpot Model 20

Figure 10. Effect of Small Changes in Input Shock Velocity Change on Number of Drops to Failure 31

Figure 11. Spring Force in Four Regions of Interest 33

Figure 12. PMMA Beam With Mass Attached to Wood Fixture 41

Figure 13. Metal Beam With Mass Attached to Wood Fixture 42

Figure 14. Spring With Attached Mass 43

Figure 15. LaserJet Printer Mounted Onto Wood Base, Plastic Programmer Testing 44

Figure 16. LaserJet Printer Sheet Metal 45

Figure 17. LaserJet Printer Mounted Onto Wood Base, Freefall Testing 49

Figure 18. Static Force Versus Deflection Curve for PMMA Beam 55

Figure 19. DBCs for PMMA Beams 61

Figure 20. DBCs for Metal Beams 62

Figure 21. DBCs for 119 Spring 63

Figure 22. DBCs for LaserJet Printer Sheet Metal 64

CHAPTER 1

INTRODUCTION AND LITERATURE REVIEW

Fragility assessment in packaging has for years been based on modeling fragile components as linear, undamped spring/mass systems inside a rigid frame (the "product"). In Figure 1, the component consists of a weight W on a linear spring having stiffness k (lbs./in) which fails when reaching some predetermined permanent deformation, or by breaking. Failure is usually taken to be complete fracture of the component in question. According to accepted theory [1], an "input" shock pulse to the product must have a critical velocity change, ΔV_{cr}, and a critical deceleration, G_{cr}, in order to transmit the shock to the component and cause failure. Modeling components as simple linear spring/mass systems led to the development of the Damage Boundary Curve (DBC), which shows pictorially the combination of velocity change and deceleration of an input shock pulse needed to damage the component (Figure 2).

To find ΔV_{cr} and G_{cr}, controlled drop tests on at least two whole products must be done to find each parameter independently. Shock machines were developed for this purpose, and are common equipment in packaging test labs. A shock machine consists of a rigid, heavy table that can be dropped from any height onto one of two different surfaces. The type of surface determines the characteristics of the shock pulse. Short duration (about 2 ms), high G shock pulses are produced when the table is dropped onto a hard elastic surface, called "plastic programmers", and are shaped like half-sine waves. Long duration, low G pulses are produced when the table is dropped onto a piston inside a cylinder containing an inert gas, called the "gas programmers", and are trapezoidal in

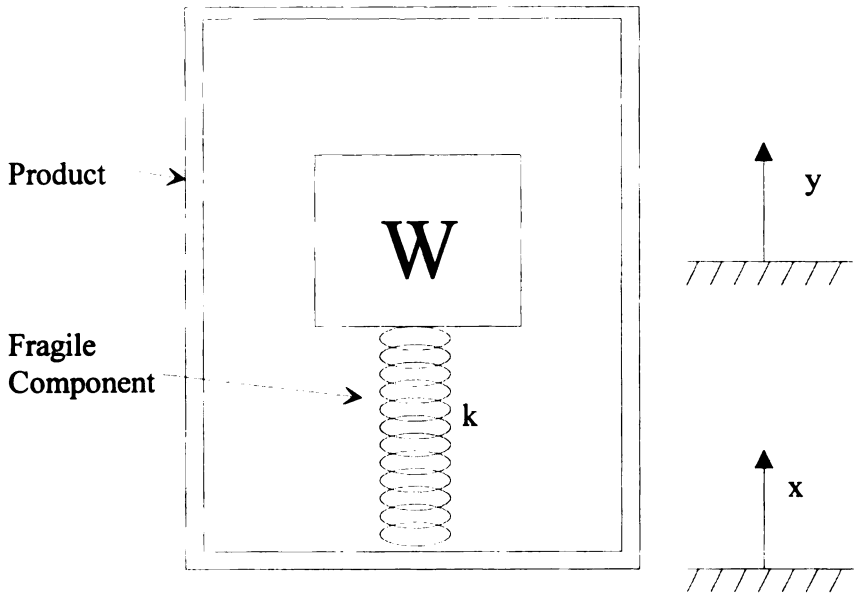

Figure 1. Product With Fragile Component, Modeled As Spring/Mass System.

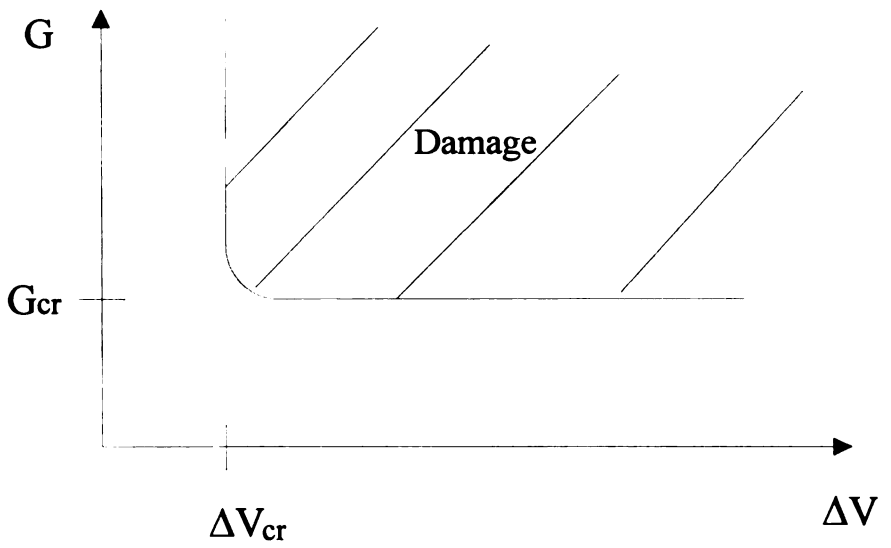

Figure 2. Traditional Damage Boundary Curve.

shape. ASTM D 3332-93 (98) "Mechanical-Shock Fragility of Products, Using Shock Machines" [2] was written to use the plastic and gas programmers to find critical velocity change and the critical deceleration values respectively for the product. ASTM D 3332 requires a sample product be fixed to the table and subjected to repeated drops of increasing intensity (increased drop height) on the plastic programmers. The critical velocity change is recorded from the input shock pulse when damage first occurs to the component. A new sample, in the same orientation as the previous one, is mounted to the table, and subjected to repeated drops of increasing intensity (increased gas pressure) on the gas programmers. The critical deceleration value is recorded from the input shock pulse that first causes damage. Often the values used for critical velocity change and critical deceleration are averages of the shock causing damage and the preceding shock. The DBC is then constructed as shown in Figure 2. The small rounded knee has a known shape [1] but is usually squared off for simplicity.

The DBC is used to evaluate the damage potential of an input shock to the product. This is done by comparing the input shock's velocity change and G level to the ΔV_{cr} and G_{cr} found on the graph. If both velocity change and G level from the input shock are greater than ΔV_{cr} and G_{cr} on the DBC, damage is predicted for that component. Since the packaging community is familiar with the DBC, ΔV_{cr} and G_{cr} will also be used in this study to eventually describe damage.

The DBC has some limitations because of assumptions used in its derivation that make its use less than ideal. The common DBC shown in Figure 2 really only applies to square shaped inputs. This is because G_{cr} is defined by a square shaped input pulse, which was used because it is more severe than other waveforms. At best then, the DBC

generated with a square shaped input is a very conservative description of failure (the DBC for it envelopes other input pulse shapes), leading to overprotection and more costly packaging solutions in many cases. Another limitation of the traditional DBC is its reliance on an expensive shock table for generation. Generally speaking, it is difficult to create and control square shaped shock pulses, even with the aid of specialized machinery. The actual shock is trapezoidal at best. Developing DBCs then relies on equipment that must be bought or rented, usually at significant cost. The traditional DBC spelled out in ASTM D 3332 also requires at least two whole products to be damaged for each orientation of testing. Testing for all six orientations can require a minimum of 12 products – an impossible requirement especially for high priced, schedule-driven products in the electronics industry. The use of many products can be quite expensive, and only data relating to velocity change and deceleration is captured. DBCs are also sensitive to the faired G values of the trapezoid shock pulses, resulting from equipment limitations and high frequency noise superimposed on the pulse (ringing). Non-trapezoidal shocks would cause the critical acceleration line of damage to vary widely as a function of velocity change. An even more serious limitation is the ignored effect of all the previous impacts leading up to damage, in both the plastic and gas programmer parts. This will be discussed later.

In the early 90's, the concept of Shock Response Spectrum (SRS) was borrowed from other engineering disciplines and applied to fragility issues in packaging [3-6]. SRS takes any input shock pulse and predicts the response of an ideal spring mass system with a known natural frequency. This is a departure from the traditional method, which focused on information about the input shock to the product, not the response of the

component. Commercial software packages generate plots that show peak G response of a component to any input shock to the product, which is a function of only its natural frequency. These plots usually range from 3 Hz to about 10,000 Hz (Figure 3). They can also calculate only the entire time response for a single component with a particular natural frequency; this is simply called the "Shock Response" (Figure 4). Many commercial software packages are available that perform this analysis, relying on the speed and power of modern computers to quickly execute the complex algorithm. This approach eliminated some of the difficulties inherent with traditional testing, including "ringing" of input shock pulses, and the frequent difficulty in measuring the response of a critical element, which is often very small and delicate.

Newton showed the relationship between SRS and the DBC, and proposed a procedure for determining fragility using shock response to trapezoidal input pulses [1]. Work was also done to use SRS in establishing shock response tolerance limits with free fall drop testing [5].

To simplify data gathering for DBCs, SRS can be used, eliminating the need for testing whole products. Using only one product and any input shock that causes damage, it has been shown that it is possible to extract the same ΔV_{cr} and G_{cr} using SRS as traditional DBC testing [6]. Deflection damage criteria was also used in conjunction with SRS, but only with the linear, elastic model inherent in the SRS algorithm. Assuming the component fails when the spring deflection reaches some critical amount, d, and assuming an ideal linear, undamped spring/mass system, the relation between component properties and whole product fragility parameters (ΔV_{cr}, G_{cr}) is:

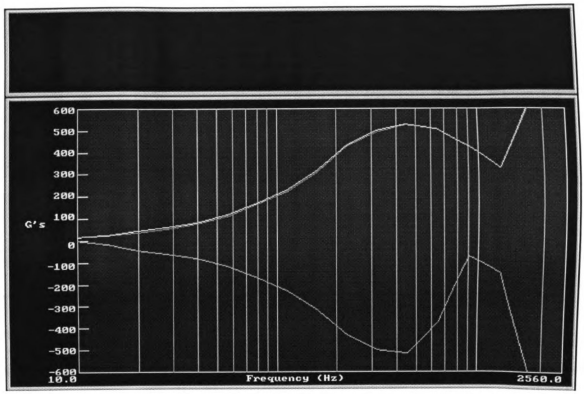

Figure 3. SRS Plot From Commercial Software By Lansmont Corporation.

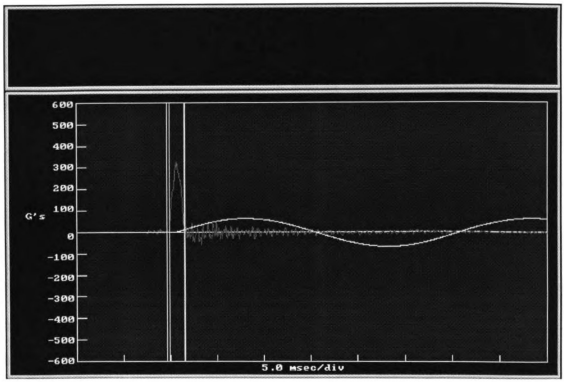

Figure 4. Shock Response Plot From Commercial Software By Lansmont Corporation.

$$\Delta V_{cr} = 2\pi f_n d \tag{1}$$

$$G_{cr} = \frac{(2\pi f_n)^2 d}{2g} \tag{2}$$

$$f_n = \frac{1}{2\pi}\sqrt{\frac{kg}{W}} \tag{3}$$

where f_n is the natural frequency of the component. The utility in using SRS for DBC generation is that if the component properties f_n and d are known, the DBC can be constructed without the need to destroy any products. Even if d is unknown, only one product need be damaged to get it, since f_n can usually be found non-destructively by conducting a resonance search on a vibration table [2] or by simply observing the vibration of the component using an accelerometer attached to the product. All of this can be done without the aid of a shock table, resulting in a streamlined and economical method for generating DBCs.

The most serious limitation in traditional DBC and SRS-generated DBC curves is the assumption that failure occurs in a brittle mode, ignoring the possibility of fatigue failure, since all shocks before damage occurs are considered to have had no effect on the product. In reality, however, most products are not brittle in nature, but ductile. ASTM D 3332 assumes a brittle model, but also does describe an alteration to the basic method called the "staircase" method, which attempts to account for the effect of multiple shocks. Several products are tested sequentially. The first product is tested at a level near its estimated failure point, and subsequent products are tested at levels higher or lower than

the previous one, depending on if the product fails or not. The average ΔV_{cr} and G_{cr} are then calculated. The data extracted from this test requires many more whole products to be destroyed than the traditional method and yet still only yields G_{cr} and ΔV_{cr} required for one drop. Recognition is also given to the premise of ductile failure in ASTM D775-80 (86), "Drop Test for Loaded Boxes" [2], which describes the number of drops from a particular height to cause failure. However, the only data extracted is number of drops versus drop height, and is specific to that test specimen and its package system. Nothing can be said of the inherent properties of the test specimen itself. The motivation for this study is the recognition that most materials are ductile in nature, not simply brittle, and a more complete accounting needs to be made between ΔV_{cr}, G_{cr} and number of drops to failure.

The brittle model assumes a linear, spring/mass/dashpot model of an internal component which fails at the elastic limit of the spring. Glass and tempered steel are examples of materials that generally fail in this mode. However, this model does not adequately describe most plastic and soft metal components encountered in many of today's electronic and home appliance products. Many products are manufactured with materials that exhibit plastic deformation (ductile in nature) and so are susceptible to fatigue damage. Soft steels, aluminum and many plastics behave this way. Simply bending these materials back and forth will demonstrate the principle. For an ideal ductile component, as shown in Figure 5, force is proportional to displacement (compression of material) up to the elastic limit (marked EL), and compression occurs under constant force after that point up to the break point (BP). This is a much different model than a simple linear spring/mass system, where the elastic limit and break point are

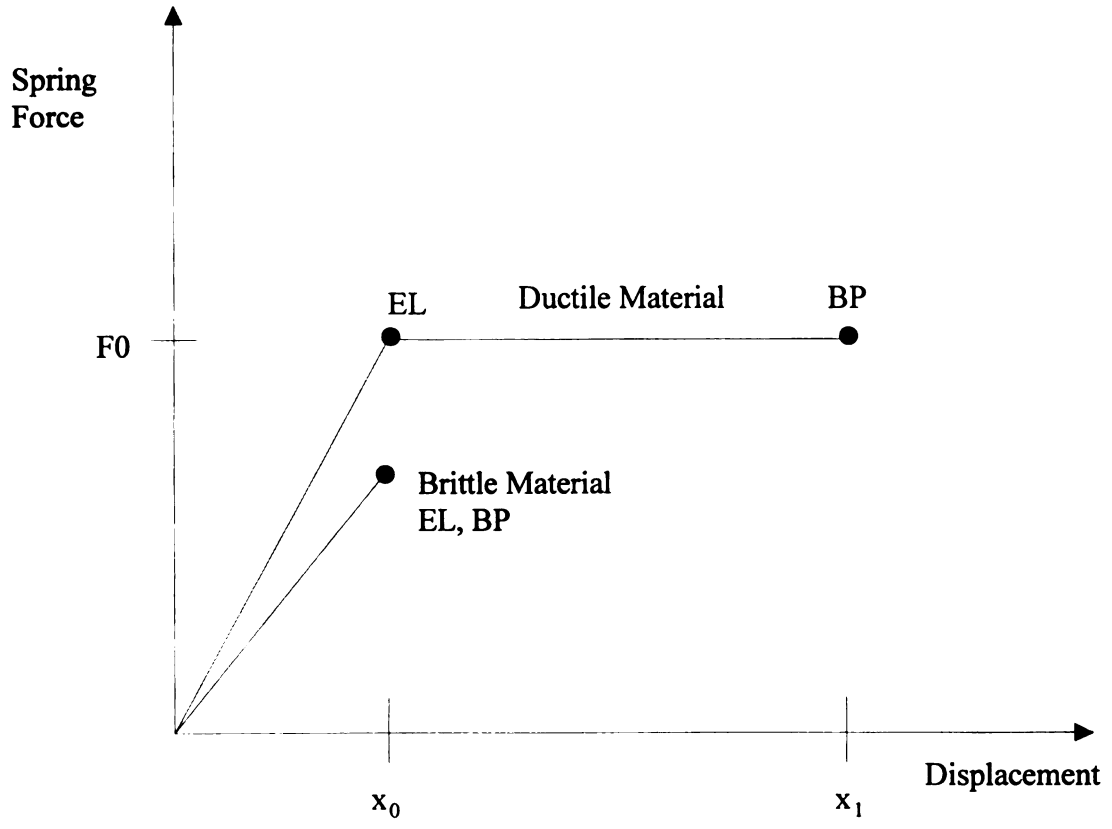

Figure 5. Dynamic Force Deflection Curve, Including Damping.

the same. Burgess recognized the need for fragility testing to include ductile failure modes, and has proposed rigid, perfectly plastic, and elastic, perfectly plastic models for components [7,8], which led to the conclusion that the number of drops to failure is also important.

The purpose of this study is to develop a mathematical model expanding the SRS technique to account for plastic deformation and fatigue damage based on a simplified version of the Bauschinger effect [9] and an idealization of the stress-strain curve (see Figure 5), with the proposal that this simplified description is sufficient for a wide range of common manufacturing materials in use today. The idealized stress-strain curve in Figure 5 is usually understood in the context of static testing conditions, and behavior of several materials have been experimentally verified to be approximated by this model [10]. This study will make use of this same idealized stress-strain relationship, but in the context of dynamic conditions. This distinction is important, since the material properties in the model will be generated and used in dynamic conditions, the normal environment for package/product damage. Components exhibiting brittle behavior can also be handled with this model because the brittle stress-strain curve can be considered to be a special case of the elastic, perfectly plastic curve: one where EL and BP are the same. If a component cannot be described as brittle, lab testing using the DBC or conventional SRS concepts can yield perplexing results. For instance, a product may not fail at a certain drop height one time, but does fail at the same level the next. For a truly brittle product, this should not happen. A component that is described as brittle can be subjected to repeated shocks without damage, providing the shock input level is below a critical value. The first shock above this level will cause damage, regardless of the number of previous

shocks below this level. An example is a rubber ball hitting a glass window. Only when the force of impact exceeds the critical resisting force of the window will there be damage, regardless of the number of previous hits below this level. For the same rubber ball thrown against an aluminum garage door, a small dent would occur on the first throw, and continue to enlarge with repeated throws.

This study is an enhancement to the traditional SRS approach for determining product fragility because it includes a method for predicting component failure due to repetitive shocks. Important material properties describing the elastic/perfectly plastic deformation behavior of the component are extracted from initial tests, and are used to predict not only the deceleration response to transient input shocks, but fatigue deflection as well. This idea of fatigue deflection centers around the traditional S-N curve [11], which shows number of cycles to failure versus force applied. The result may be considered a new approach for constructing DBCs using SRS, which includes not only velocity change and deceleration information, but number of cycles to failure as well.

Traditional DBC generation requires a shock machine with gas and plastic programmers. However, using SRS and a ductile model, all relevant information for constructing a robust fatigue DBC can be extracted using only the plastic programmers, or even more significantly any input shock, such as those generated from freefall drops. Using the component's material properties, the component's response to shock pulses from any source can be evaluated in terms of deceleration and fatigue damage (displacement).

The software developed in this study ("PROGRAM") allows the user to input the component material properties and a digitized shock pulse, then calculates and displays

the shock response, predicted percent damage, and the predicted number of identical drops to cause failure (failure pre-defined by the user). Several commercial software packages exist that are able to produce Shock Response Spectrums, given an analog shock input. However, there are none known that incorporate a fatigue model.

There are three important parameters needed to describe ductile damage, as shown in Figure 6. The displacement x_0 and spring force F_0 denote the elastic limit, the point corresponding to the end of a proportional relationship between applied force and displacement. The displacement x_1 denotes the end of the perfectly plastic region, and corresponds to failure of the material. Again, the model in this study is assumed to describe material behavior under dynamic loading conditions.

Stretching a ductile material up to the elastic limit and releasing it results in complete recovery of the material to its original size and shape, and the material returns to zero displacement. Stretching the ductile material past the elastic limit but short of the break point results in permanent deformation when the load is completely removed. As the load on the material is being released, the force versus displacement relationship follows a path parallel to the elastic loading line, designated as elastic unloading, as shown in Figure 7 [9]. As a result, it does not return to its original length, but has some permanent displacement corresponding to the point where the unloading line crosses the displacement axis. Multiple stretching and releasing of the ductile material past the elastic limit gives a force/deflection history as shown in Figure 7, following path 0-1-2-3 in the first cycle, 3-4-5-6 in the second cycle, 6-7-8-9 in the third cycle, and 9-10-11 in the fourth. At point 11 in this example, the break point is reached, and the material fails. For a product being dropped, this would correspond to failure after the fourth drop,

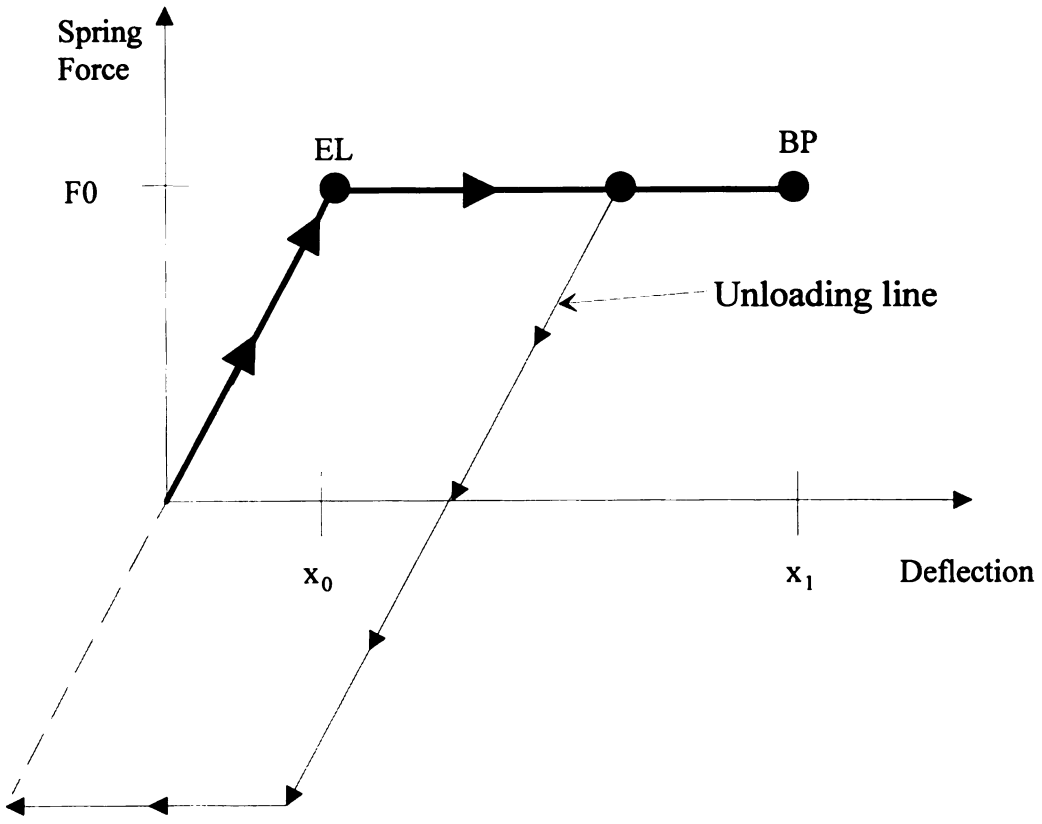

Figure 6. Material Parameters Describing Ductile Damage.

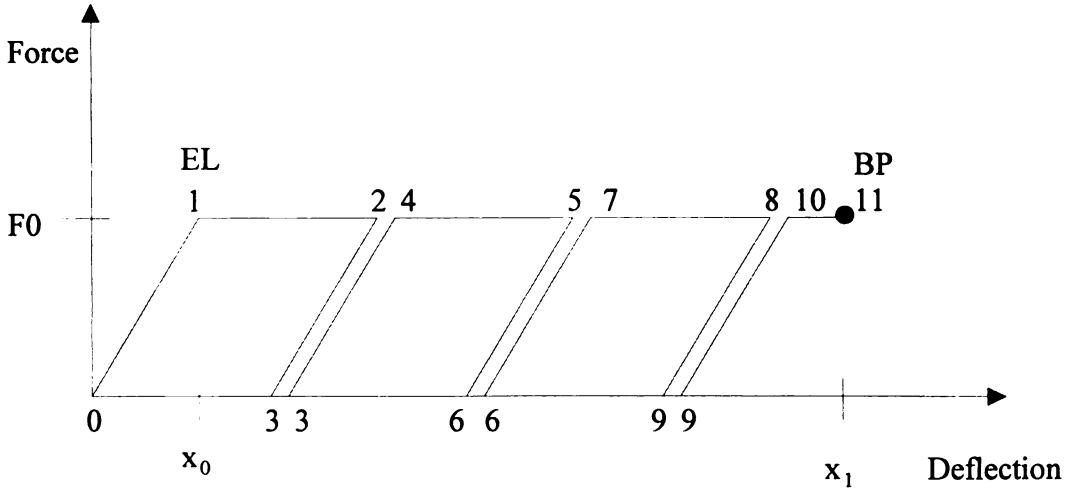

Figure 7. Multiple Stretching and Releasing of a Ductile Component.

assuming each input shock pulse was identical. A large enough drop could cause the component to follow the path 0-1-11, and break in one drop. This theory of damage accumulation is based on a linear damage rule, also known as the Palmgren-Miner hypothesis [11]. This rule is widely used in the generation of S-N curves because of its simplicity and the experimental fact that other much more complex cumulative damage theories do not always yield a significant improvement in failure prediction reliability.

The elastic, perfectly plastic stress-strain curve in Figure 6 can be used to modify the traditional DBC by accounting for fatigue through the number of drops to failure [8]. Using the plastic and gas programmers, critical velocity change, critical deceleration and number of drops to failure are related by:

$$\Delta V_{cr} = \sqrt{2g\left(A + \frac{B}{N}\right)} \tag{4}$$

$$G_{cr} = \omega\sqrt{\frac{2A}{g}}\left(\frac{A + B/N}{2A + B/N}\right) \tag{5}$$

where:

$$A = \frac{\omega^2 x_0^2}{2g} \tag{6}$$

$$B = \frac{\omega^2 x_0^2}{g}\left(\frac{x_1}{x_0} - 1\right) \tag{7}$$

$$\omega = 2\pi f_n = \sqrt{\frac{F_0 g}{x_0 W}} \tag{8}$$

and

$g = 386.4$ in/sec^2 (gravity)

W = component weight

N = Number of drops to failure

x_0, x_1 and F_0 are spring properties, as shown in Figure 6

Figure 8 shows generalized DBC's for a product with certain A and B values. Now critical velocity change and critical deceleration will depend on the number of drops to failure (N).

It is important to point out that the spring properties when found are interpreted as dynamic properties. Obtaining x_0 and x_1 from dynamic testing automatically accounts for another material property, damping. This means that natural frequency in Equation 8, which normally defines natural frequency for a linear undamped spring/mass system, now relates to a linear damped spring/mass system due to its relationship to dynamic properties F_0 and x_0. The results in Equations 4 through 8 will be used as a check against the methods developed for SRS-generated DBCs in this study. The brittle model results (Equations 1 and 2) can be recovered from the above equations by putting $x_1 = x_0$. When $x_0 = 0$, the rigid, perfectly plastic model is recovered (see Appendix A). As B in Equations 4 and 5 approaches zero, the effect of N on ΔV_{cr} and G_{cr} is reduced, and so a large number for A or a small number for B indicates a more brittle component. Conversely, a small value for A or a large value for B indicates a more ductile component. A and B may be regarded as two material properties associated with the

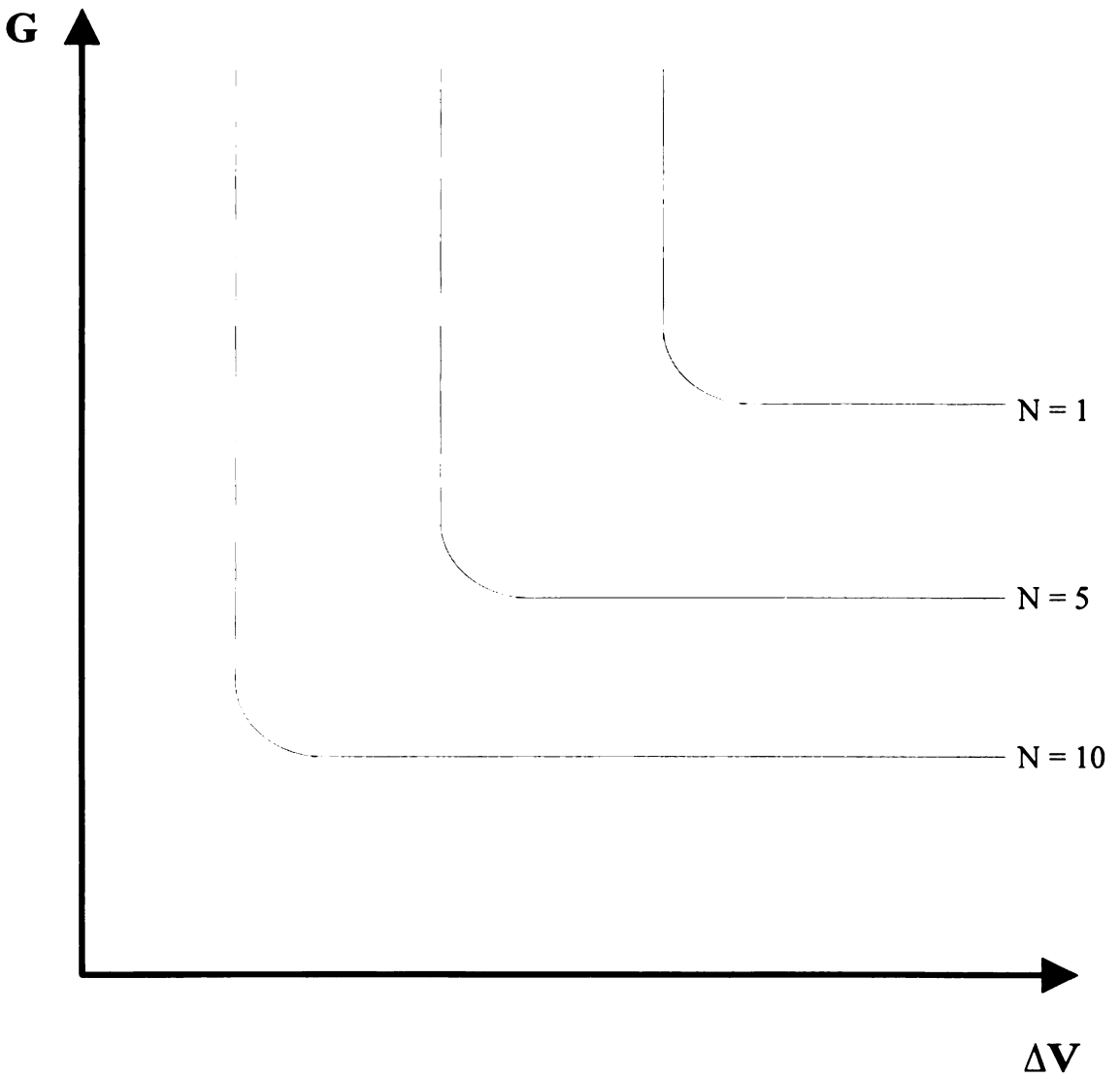

Figure 8. Generalized DBCs Incorporating Fatigue Damage.

component that replace x_0, x_1 and F_0. This allows for simpler expressions describing ΔV_{cr} and G_{cr}. The use of A and B also helps to point out the relationship between fatigue theory and the observations of ASTM D 775. To show this relationship, recall velocity change from a plastic programmer drop is equivalent to impact velocity in a free fall drop [12]. Thus,

$$\Delta V_{cr} = \Delta V_{table} = \sqrt{2gh} \tag{9}$$

Substituting into Equation 4 and solving for the equivalent free fall drop height, h, yields:

$$h = A + \frac{B}{N} \tag{10}$$

Plotting Equation 10 gives the hyperbola-shaped relationship describing drop height and number of drops to failure observed in ASTM D 775. A and B therefore have physical meaning related to free fall drop height. As number of drops to failure gets large, drop height approaches A, and so A is the smallest drop height required for failure. When number of drops to failure is one, drop height is A plus B.

An inherent limitation associated with predicting material failure due to fatigue are variations in the material itself. It is widely recognized that scatter in material properties is common, and can be wide, when describing damage from stress-strain applications [13]. Statistical methods are necessary to account for these material variabilities [10,13]. These issues will be addressed later in this study.

Finally, some early data exists showing qualitatively the effect of number of drops on damage and generation of DBCs. Singh [14] used light bulb filaments and bricks to experimentally demonstrate that DBCs can be generated for different number of drops causing failure at different energy inputs (shock table velocity change). It is interesting to note that bricks appeared to follow the same pattern, an unexpected result since bricks would normally be considered brittle. The implication is that a fatigue model might be a good fit for crack propagation failure. In any case, no physical explanation was given for the observations, but the results do lend support for the premise of ductile failure and the incorporation of that information into DBCs.

CHAPTER 2

THEORETICAL DEVELOPMENT

2.1 FINDING MATERIAL PROPERTIES x_0, x_1 AND F_0

The first objective when defining fragility using an elastic/perfectly plastic model is to determine the material properties x_0 and x_1, which correspond to the elastic limit and break point, respectively (See Figure 6). Intuition suggests finding x_0 and x_1 from a simple static force/deflection test. However, practice has shown that viscoelastic materials can behave quite differently under dynamic rather than static loads [15]. In drop situations, events causing damage occur in dynamic modes, so care must be taken to determine x_0 and x_1 accordingly. Also, recall that the ductile model in Figure 6 is assumed to apply to dynamic, not static conditions. To demonstrate how material behavior can be quite different under dynamic versus static conditions, consider a simple spring/mass/dashpot model under static loading (Figure 9). The force exerted on the mass is $F = kx + c\dot{x}$, where k is the spring constant and $\dot{x}$ is velocity. In a static compression test, only the spring force kx is active and in a dynamic test, both the spring force and the dashpot force $c\dot{x}$ are present. The dashpot force can be considerable, giving very different results. In view of this, x_0 and x_1 can be found experimentally from shock tests using both gas and plastic programmers [8]. There are two other methods that may also be used to find x_0 and x_1, each with advantages over using both gas and plastic programmers. The first is to use only the plastic programmers, eliminating the need for the gas programmers. The second method is to use freefall drop testing, eliminating the need for a shock table altogether.

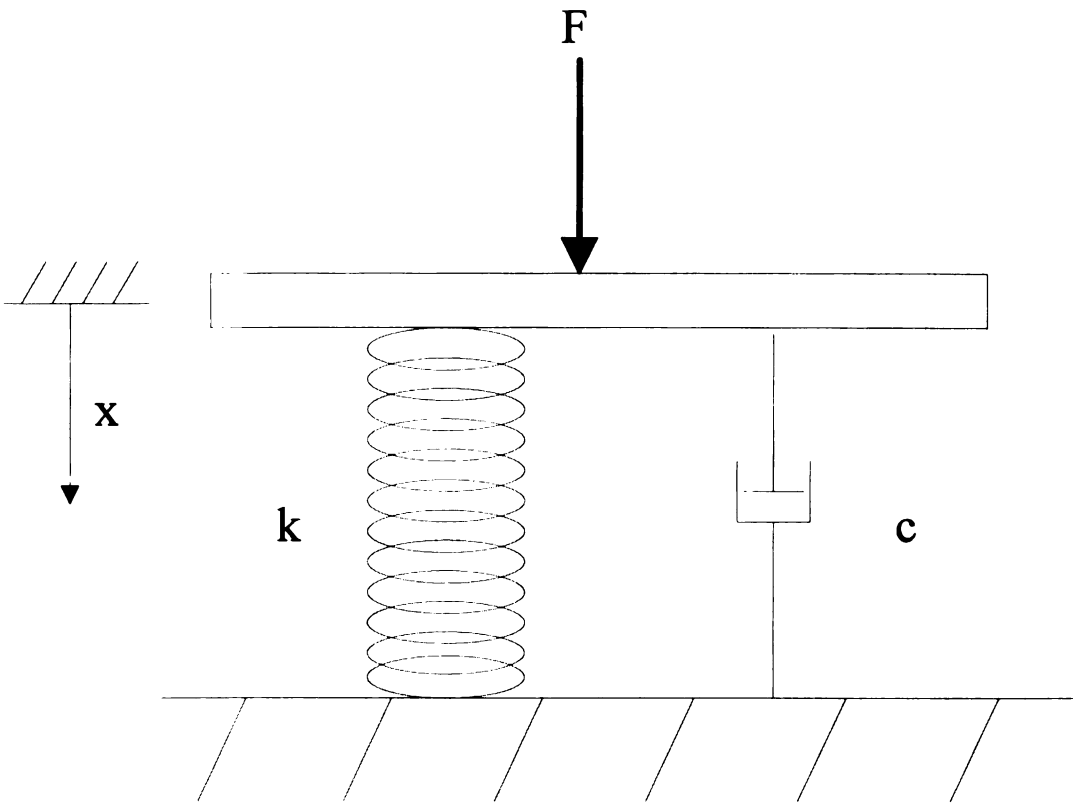

Figure 9. Simple Spring/Mass/Dashpot Model.

2.1.1 Plastic Programmer Method For Determining Material Properties

The first method, to be called the "plastic programmer method," relies on the nature of the shock produced on the plastic programmers. In general, short duration shocks of about two milliseconds can be produced. This corresponds to an "input shock pulse frequency" of $\frac{1}{2(.002\,\text{sec})} = 250$ Hz. As long as this frequency is larger than the natural frequency of the spring/mass system, which is usually the case, the peak response of the spring/mass system will occur after the input shock is over because the mass will not have had time to react during the input shock. This observation was used in the derivation of Equation 4. In practice, the input shock pulse frequency need be only about two times larger than the spring/mass natural frequency to make this statement [12]. For a two millisecond half sine pulse, this would apply to spring/mass components of 125 Hz or less. This covers many products.

To find x_0 and x_1 using the plastic programmer method, the material properties A and B in Equation 4 should first be found. A suggested procedure is as follows:

1. Place a new product on the shock table and set the table for a low level drop. Raise and drop the table, recording the velocity change. If no damage is found, repeat the drop at the same height until failure occurs, keeping track of the number of drops. This number will be known as the number of drops to failure, N, at this velocity change. It is important to note there may be some ambiguity as to what number to use for N when damage does occur. Since damage accumulates, choosing N to be the last drop when failure is observed is too high and choosing N to be one drop less than this is too low. Thus, N is probably somewhere between the two drops. We must make some rationale as to which N to use. Choosing the drop before damage is

observed clearly is not a good strategy since the objective is to reach failure. A reasonable approach might be to average the number of drops (use 4.5 when damage occurs on the 5th drop). However, since number of drops to failure in a practical sense is limited to integer values (we only observe the 4th and 5th whole drops), and since damage is the clear objective, choosing the drop when failure is observed provides a straightforward albeit "conservative" method. For accurate results, this low level dropping should yield many drops, maybe 20, before failure. It is likely ΔV will vary slightly from drop to drop, and therefore the average ΔV and corresponding N becomes the first data point (ΔV_{avg}, N). Because of material variability, it is recommended to record ΔV and N for several units at each table height, and to use average ΔV and average N for generating the data point.

2. Having established damage at a particular height (ΔV) from Step 1, take a new product and mount it on the table. Choose a higher drop height from which to drop. Proceed as before, dropping the table repeatedly and recording ΔV and the number of drops to failure. This will establish failure at a higher energy input. For this drop height, the average ΔV and number of drops to failure become the second data point. At least two data points are needed, since there are two unknowns, A and B. This step may be repeated for different drop heights, establishing other data points, in which case A and B will be found using regression over all the data points.
3. To find A and B, first convert all of the plastic programmer velocity change data into equivalent free fall drop heights, $h_{equivalent}$, using Equation 9 cast as:

$$h_{equivalent} = \frac{\Delta V_{table}^2}{2g} \tag{11}$$

Doing so will give the data points expressed as pairs of h and N rather than ΔV and N. Equation 10 now expresses a linear fit between h and 1/N which can be solved using standard regression techniques as follows. The discrepancy between the true value of h and the predicted value A + B/N is

$$h - A - \frac{B}{N} = error \tag{12}$$

Minimizing the sum of the squares of the residuals (errors) for "m" data points

$$SSE = \sum_{i=1}^{m}\left(h - A - \frac{B}{N}\right)^2 \tag{13}$$

requires that

$$\frac{\partial}{\partial A}(SSE) = 2\sum\left(h - A - \frac{B}{N}\right)(1) = 0 \tag{14}$$

$$\frac{\partial}{\partial B}(SSE) = 2\sum\left(h - A - \frac{B}{N}\right)\left(\frac{1}{N}\right) = 0 \tag{15}$$

Solving simultaneously for A and B gives:

$$A = \frac{\sum \frac{1}{N^2} \sum h - \sum \frac{1}{N} \sum \frac{h}{N}}{m \sum \frac{1}{N^2} - \left(\sum \frac{1}{N} \right)^2} \tag{16}$$

$$B = \frac{m \sum \frac{h}{N} - \sum \frac{1}{N} \sum h}{m \sum \frac{1}{N^2} - \left(\sum \frac{1}{N^2} \right)^2} \tag{17}$$

$$R = \frac{m \sum \frac{h}{N} - \sum \frac{1}{N} \sum h}{\sqrt{m \sum \frac{1}{N^2} - \left(\sum \frac{1}{N} \right)^2} \sqrt{m \sum h^2 - \left(\sum h \right)^2}} \tag{18}$$

where all summations in 14 – 18 are over the number of data points, and R is the correlation coefficient. Squaring R will give the proportion of drop height variability that is explained by the linear regression model. As R^2 approaches unity, the better the model explains the residuals.

One last piece of information is needed before finding x_0 and x_1: component natural frequency, which can be found non-destructively. A simple method is to place the unit on a vibration table and perform a sine sweep test [2] noting the frequency corresponding to resonance. If available, using an electromagnetic vibration table is recommended since table performance tends to be more accurate than hydraulic-driven tables, especially at higher frequencies. Other methods for finding natural frequency are given in previous works [6]. Now x_0, x_1 and F_0 can be found by rearranging Equations 6, 7 and 8:

$$x_0 = \sqrt{\frac{2Ag}{\omega^2}} \tag{19}$$

$$x_1 = x_0 + \frac{gB}{\omega^2 x_0} \tag{20}$$

$$F_0 = \omega^2 x_0 \frac{W}{g} \tag{21}$$

By definition, the SSE calculated in Equation 13 represents the variance in freefall drop height. Dividing by m and taking the square root gives

$$S = \sqrt{\frac{SSE}{m}} \tag{22}$$

where S is the standard deviation in the predicted equivalent free fall drop height [16]. This will be a measure of both the model error and experimental error. A small experimental error is expected, since velocity change of the shock table can generally be held to within +/- 5%.

In the minimization of SSE in Equation 13, a high number of drops to failure is weighed equally with small number of drops to failure. This may prove troublesome if the effects of material variability are pronounced. Because packages are normally dropped only a few times from heights that cause failure [17], less importance can be placed on high number of drops to failure. Greater importance can be given to small N in Step 3 of Section 2.1 by weighting residuals,

$$\sum_{i=1}^{m}\left(h - A - \frac{B}{N}\right)^2 \cdot W_i = SSE \qquad (23)$$

where $W_i = \frac{1}{N_i}$, for example. Other weighting schemes might be considered, such as taking $W_i = P_i$, the probability of dropping the package a certain number of times. This kind of data does not readily exist, making the justification for using a certain P_i no more or less arbitrary than other weighting schemes. Still, this might be a useful improvement if such data is reliable and available, or if testing in Steps 1 and 2 yields a large difference between number of drops to failure at the respective drop heights. The method chosen for this study was to use Equation 13. The reason is shown in Table 1. For each of the four models tested in this study, the observed number of drops to failure varied similarly regardless of drop height. This suggests that the materials tested behaved with similar variance under the conditions tested.

Table 1. Variation in Observed Number of Drops to Failure for Test Models.

	Table Height 1	Table Height 2	Table Height 3	Average
Model 1	49%	46%	51%	49%
Model 2	23%	12%	23%	19%
Model 3	19%	9%	N/A	14%
Model 4	50%	36%	N/A	43%

2.1.2 Freefall Method For Determining Material Properties

The second method investigated in this study was to obtain material properties x_0 and x_1 from any input shock - in this case, shocks from freefall drops. This simple method eliminates the shock table altogether, and relies solely on freefall drop testing with the test product placed in a cushioned package. The procedure for dropping and recording number of drops to failure is similar to the one described in 2.1.1, with one notable exception: the SSE to be minimized is the difference between the known number of drops to failure and the predicted number for the given input shock pulse. The predicted number of drops to failure uses the algorithm described in detail in the next section. Since the predicted number of drops requires x_0 and x_1 to be known beforehand, an educated guess on their values is made, and then the correct x_0 and x_1 are then found by an iterative method: systematically varying them until the SSE is minimized. A summary outline can be found in Appendix B. This approach must be used since the shock pulses in a typical freefall drop test usually are not short enough in duration to be considered spikes to the component, thus nullifying the relationship in Equation 10. The advantage of this approach is its simplicity: the shock table is eliminated and only freefall drops need to be done. The apparent disadvantage of not knowing x_0 and x_1 can be eliminated by estimating their starting values as in plastic programmer testing, since the values are expected to be similar. Even if nothing is known about x_0 and x_1, the program will still find the appropriate values, though this may be a lengthy process. A potential disadvantage of this method is the importance of accelerometer placement, since SRS assumes the shock it analyzes is in fact the true imput shock to the component. It

may be difficult to determine the proper accelerometer location, depending on the test model. The procedure, however, is straightforward:

1. Place an accelerometer on the frame of the product, and choose a drop height from which to drop the packaged product. Repeatedly drop the product until failure occurs, recording number of drops to failure, and capturing the shock pulses from the accelerometer. As with the plastic programmer method, there may be some ambiguity as to what number to use for N when damage occurs, but the same conclusion for choosing N is reached for freefall testing – choose N to be when failure is observed. The choice of cushion material is important, since number of drops to failure is recorded with the assumption that each shock pulse is similar: the cushion (and to a lesser extent the corrugated container) must recover most of its cushioning properties between drops. This eliminates certain cushion materials since performance degrades rapidly after the first drop. Therefore it is best to choose a high quality resilient cushion that performs consistently after multiple drops – neoprene is recommended if it is available. Of course the cushion could be replaced after each drop, but this may prove unnecessarily time consuming, and small variations in ΔV and G will still occur between drops. As in the plastic programmer method, material variability of the critical component is to be expected, so it is recommended to test more than one unit at this height, and take the average number of drops to failure as N. The number of drops to failure and corresponding input shock pulse become the first data set.
2. Having found damage at a particular freefall drop height from Step 1, take a new product in a cushioned package and choose a higher height from which to drop.

Proceed as before, dropping the package repeatedly, recording number of drops to failure. This will establish failure at a higher energy input. The number of drops to failure and corresponding input shock at this drop height becomes the second data set. Two data sets are needed since there are two unknowns x_0 and x_1 which must be solved for. This step may be repeated for different drop heights to establish other data sets.

3. A program following the outline in Appendix B can now used by entering the number of drops to failure observed from each height and representative digitized shock pulses from each height. If variability in ΔV and G between repeated drops is a concern, it is possible using existing hardware and software to reconstruct an average resultant input shock for the program, giving one representative pulse for each drop height tested. Executing the program returns the values for x_0 and x_1 corresponding to the minimized SSE, as well as the calculated values for material properties A and B. ΔV_{cr} and G_{cr} for each N is also calculated and displayed. The program could also be extended to draw the corresponding fatigue DBCs.

2.1.3 Sources of Experimental Error

There are two sources of experimental error which can account for part of the SSE of Equation 13: variation in material properties among the individual specimens tested, and fluctuations in ΔV produced by either the shock table or in freefalls during testing. Variation in the material is evident from the large variations in the number of drops to failure shown in Table 1. Traditional S/N curves readily acknowledge the problem of scatter in collected data [10,13], and materials can exhibit a range of values during testing

for many reasons. For high molecular weight plastics, engineering thermoplastics and thermosets, molecular weight distribution is especially critical. Even small variations can lead to noticeable performance changes [18]. Impurities found in the material can also create voids, weakening some samples, or providing additional reinforcement. Residual stresses, from mechanical fabrication or thermal or mechanical treatment can also cause variation. Number of drops to failure will be affected by these factors. Material anisotropy is also a concern, especially when parts can be manufactured randomly with respect to the property directions.

Small changes in ΔV can also affect number of drops to failure, but usually only for large N. As an illustration, consider Figure 10. If the input shock puts the compression of the spring just past the elastic limit at point A, the number of drops to failure is $\frac{x_1 - x_0}{\varepsilon}$. If the input shock is changed only slightly, it is conceivable the compression of the spring can be pushed further past point A, say 2ε. Then the number of drops to failure is $\frac{x_1 - x_0}{2\varepsilon}$, half as many as before.

2.2 SRS AND DUCTILE MODEL

Having found x_0 and x_1, the ductile model can now be added to the SRS algorithm, providing a means for predicting fatigue failure in response to any input shock. Applying Newton's second law to the spring/mass system in Figure 1 gives:

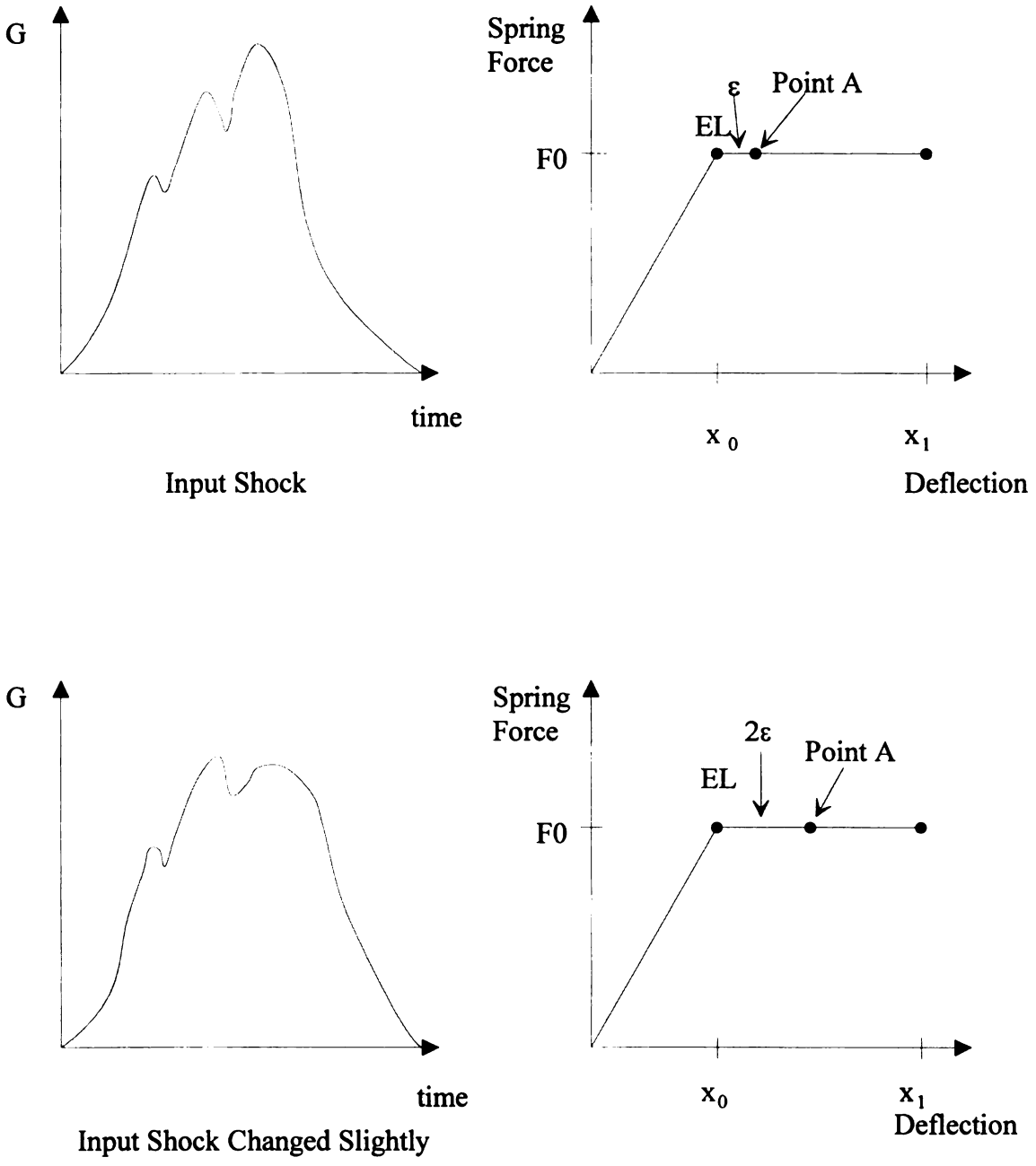

Figure 10. Effect of Small Changes in Input Shock Velocity Change on Number of Drops to Failure.

$$F = m\ddot{y} \tag{24}$$

where F is the spring compression force and m = W/g is the mass. Damping is not explicitly included in Equation 24 since the properties in Equations 19 through 21 are obtained under dynamic test conditions. As such, F depends only on the spring compression z, which in Figure 1 is:

$$z = x - y \tag{25}$$

where x(t) and y(t) are the positions of the base of the product and the component relative to their static positions, respectively, and the dot notation in Equation 24 indicates derivatives with respect to time. Substituting for y from Equation 25, and noting that $F = F(z)$, Equation 24 reads

$$\ddot{z} = \ddot{x} - g\frac{F(z)}{W} \tag{26}$$

It is advantageous to use Equation 26 over Equation 24 since it involves the acceleration of the product $\ddot{x}$, which is easily obtained experimentally using an accelerometer.

The spring force in Equation 26 changes depending where on the curve in Figure 11 one is at. There are four regions of interest. Region 1 is the elastic loading region. A linear relationship is assumed between spring force and displacement so that

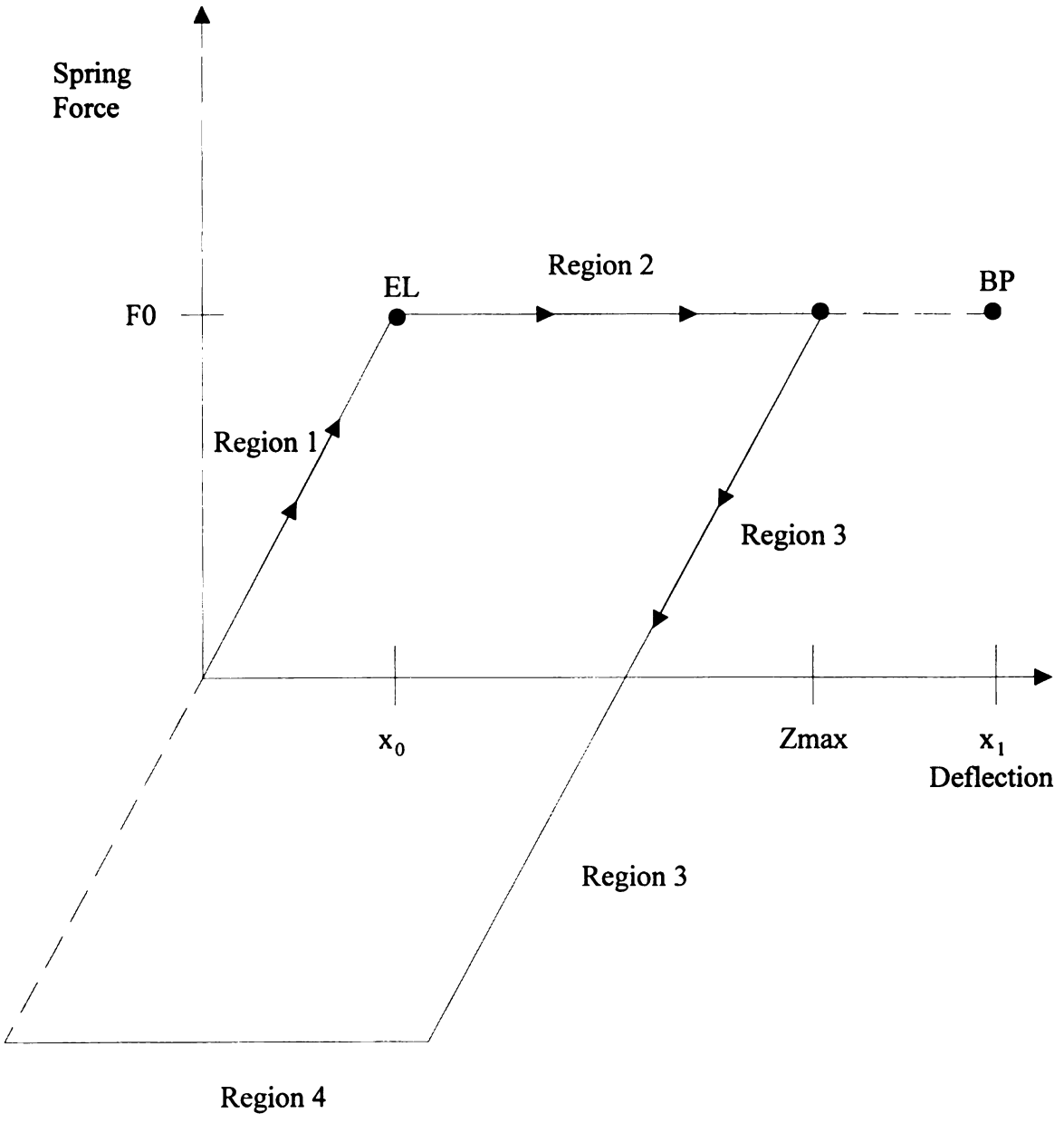

Figure 11. Spring Force in Four Regions of Interest.

$$F(z) = kz \qquad \text{if } z \le x_0 \tag{27}$$

where k is the spring constant, as shown in Figure 1. The spring force will be described by Equation 27 as long as z is less than or equal to x_0. Region 2 is the perfectly plastic region, and is described by a constant spring force

$$F(z) = F_0 \qquad \text{if } x_0 \le z \le x_1 \tag{28}$$

Equation 28 is valid if z is greater than x_0. In a dynamic situation, the spring force remains constant until $\dot{z}$ reaches zero, indicating the onset of unloading and corresponding to the maximum deflection z_{max}. Assuming spring compression does not reach the break point x_1, Region 3 describes elastic unloading, parallel to the elastic region. The spring force in this elastic unloading region is described by

$$F(z) = F_0 - k(z_{max} - z) \tag{29}$$

Region 4 is similar to Region 2, and is known as the plastic unloading region, if it even occurs during a normal shock.

If the compression of the spring ever exceeds x_1, failure occurs. If compression of the spring is below x_0, no permanent damage occurs. If compression is between x_0 and x_1, damage will be the fraction $\frac{z_{max} - x_0}{x_1 - x_0}$, accumulating linearly. By providing any input shock, $\ddot{x}$, the response $\ddot{z}$ can be calculated using Equation 26 and z can be propagated in

time using concepts developed in the next section. Now both peak G and displacement can be determined and used to assess damage potential for both brittle and ductile components.

As in traditional SRS, the reaction of the mass during the input shock duration is known as the primary response, and the reaction of the mass after the duration of the input shock pulse is known as the residual response. The larger of the two responses is reported as the maximum shock response deceleration value.

2.3 NUMERICAL SOLUTION

The equations above for finding deceleration and displacement must be solved numerically on a computer since $\ddot{x}$ will be provided at discrete points in time from a digitized shock pulse. Since instantaneous spring compression z is necessary in order to evaluate the spring force and hence $\ddot{z}$, a time step approach was used to solve the equations of motion. From elementary physics [19], if the acceleration $\ddot{z}$ at time t is assumed to be constant over the time step Δt, and $z(t)$ and $\dot{z}(t)$ are the initial displacement and velocity respectively, the spring compression z and rate of compression $\dot{z}$ at the next time (t + Δt) are:

$$z(t+\Delta t) = z(t) + \dot{z}(t)\Delta t + \frac{1}{2}\ddot{z}(t)\Delta t^2 \tag{30}$$

$$\dot{z}(t+\Delta t) = \dot{z}(t) + \ddot{z}(t)\Delta t \tag{31}$$

When working with digitized shock pulses, it is customary to use a "sampling frequency", which denotes the rate at which the computer picks off instantaneous values from the shock pulse to use in evaluation. Since $\ddot{x}$ is known, and Δt is obtained from the reciprocal of the sampling frequency, the output deceleration $\ddot{z}$ can be calculated by substituting Equations 30 and 31 into Equation 26 and solving. The program for this algorithm (PROGRAM) was written in Visual Basic 5.0, and can be found in Appendix C.

2.4 GENERATING FATIGUE DBC CURVES

Generating DBCs incorporating fatigue requires three pieces of information: ΔV_{cr}, number of drops to failure, and G_{cr}. Using the plastic programmer method, number of drops to failure and ΔV_{cr} will have already been defined in Section 2.1, Steps one through three. G_{cr} can then be found using Equation 5. Using the freefall method, number of drops to failure has already been observed, and ΔV_{cr} and G_{cr} can be calculated from Equations 4 and 5. The data is plotted similarly as shown in Figure 8.

2.5 TEST TO VERIFY PROGRAM ALGORITHM

To verify that the model works properly, a hand check was done to compare the results from two pulse shapes easy to evaluate by hand. The pulse shapes are a square wave and a half sine pulse. The solution of Equation 24 for the square pulse ($\ddot{x} = Gg$) is:

$$z(t) = \frac{GW}{k} \tag{33}$$

which satisfies the initial conditions $z = \dot{z} = 0$. The solution to Equation 24 satisfying initial conditions $z = \dot{z} = 0$ for the half-sine pulse ($\ddot{x} = Gg \sin \omega t$) is:

$$x(t) = \frac{Gg}{\frac{gk}{W} - \omega^2} \sin \omega t \qquad (34)$$

where G is peak G of the half-sine pulse. 33,333 Hz was used for the sampling frequency to force the PROGRAM to interpolate many values for the input shock $\ddot{x}$. Table 2a summarizes the results for a 20 G, 20 millisecond square pulse. Table 2b summarizes the results for a half-sine pulse with a peak G of 200 and a duration of 2 milliseconds. Only three points were chosen from the square pulse and 9 from the half-sine, enough to adequately describe the pulses. Natural frequency used was 17 Hz, and the values for x_0 and x_1 were taken to be .787 inches and 1.93 inches, respectively. These values are representative of small plastic beams with a mass attached to one end. The results in Tables 2a and 2b show good agreement. The small discrepancies can be attributed to computer round-off error and the interpolation between the relatively few data points given to the program for describing the pulses.

Table 2a. Comparison of Values, Trapezoid Pulse.

Spring		Hand Check	Computer
At Elastic Limit	Time (s)	.01626	.01626
	Position (in)	.78700	.78783
	Velocity (in/sec)	71.51514	71.49242
At End of Input Shock	Time (s)	.02000	.02004
	Position (in)	1.04900	1.04913
	Velocity (in/sec)	66.76000	66.73914
At End of Plastic Loading	Time (s)	.02748	.02748
	Position (in)	1.29760	1.29869
	Velocity (in/sec)	0.00000	.50099

Table 2b. Comparison of Values, Half-Sine Input Pulse.

Spring		Hand Check	Computer
At End of Input Pulse	Time (s)	.00200	.00201
	Position (in)	.09751	.09635
	Velocity (in/sec)	97.29128	96.48929
At Elastic Limit	Time (s)	.01070	.01071
	Position (in)	.78710	.78193
	Velocity (in/sec)	48.91088	49.59233

CHAPTER 3

MATERIALS AND TEST METHODS

3.1 MATERIALS

To capture analog shock inputs, software and hardware from Lansmont Corporation called Test Partner 2 (TP2), were used. TP2 converts analog shock pulses captured from accelerometers into digital data. For this study, the shock pulses were translated into a .csv format to be imported by the Program. Equipment used in this study included the following:

Calipers:	Mitutoyo, Model CD-6" BS, serial number 0010283
Accelerometers:	PCB Triax, Model 356A11, serial numbers 4201, 4202
Charge Amplifiers:	PCB Model 482A16
Shock Table:	Lansmont Model 65/81, s/n 57-681-0016
Vibration Table:	Lansmont Touchtest System
	Model 10000-10, size 152 cm, Hydraulic power supply
Drop Tester:	Lansmont Model PDT-56E Freefall Tester

Models

Four different spring/mass models were used for testing, and are described below:

1. PMMA beams. These beams were constructed of polymethylmethacrylate (PMMA) material, commonly known as Plexiglass^{Tm}. The material was obtained and cut into 6 5/8" x 1 3/8" x 1/4" pieces. A 9/32" hole was drilled on either end. To one end a steel mass was attached, and the opposite end was attached to a wood test fixture, as shown in Figure 12.

2. Metal Beams. Two-inch zinc-plated steel mending plates were mounted to a fixture as shown in Figure 13. A mass consisting of 3 separate brass pieces was attached to the opposite end of the beam using a small bolt and wing nut. Deflection measurements were measured from the center of mass, an assumption used in Equations 24 and 26 relating to Newton's second law.
3. Springs. As shown in Figure 14, a small spring was suspended from a long bolt and held in place using a series of nylon spacers and wing nuts. A lead-filled aluminum mass was attached to the opposite end of the spring. The springs were stock items from the local hardware store, known as 119 Springs.
4. Sheet Metal. Hewlett Packard 5L LaserJet printers were used as a product test. Removing the plastic housing and laser unit, the product was mounted onto a wood base as shown in Figure 15. The sheet metal on the bottom side was the area of interest, specifically the area shown in Figure 16.

3.2 METHODS

3.2.1 Determination of Natural Frequency

To find natural frequency of each model, two methods were employed. The first was to place the test specimen on the vibration table and watch for resonance as the table swept from 3 to 300 Hertz [2]. All four models were tested this way. The second method was to mount an accelerometer onto the component, capture the oscillations after setting it into motion, and calculating natural frequency from the response. All models except the sheet metal were tested in this manner. The sheet metal proved to be difficult to analyze, due to multiple resonant frequencies and noise superimposed on the

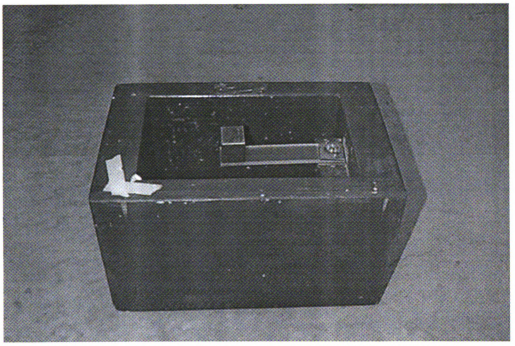

Figure 12. PMMA Beam With Mass Attached to Wood Fixture.

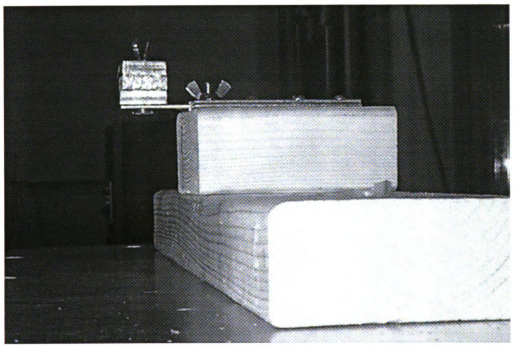

Figure 13. Metal Beam With Mass Attached to Wood Fixture.

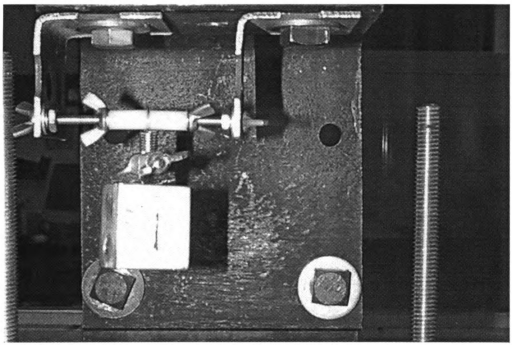

Figure 14. Spring With Attached Mass.

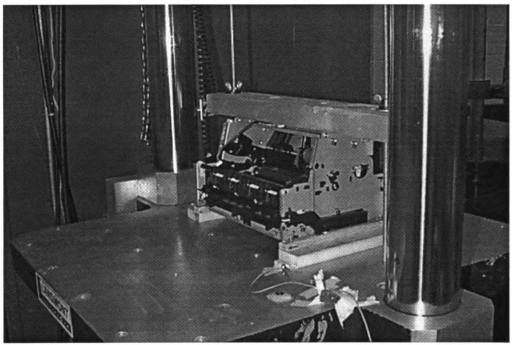

Figure 15. LaserJet Printer Mounted Onto Wood Base, Plastic Programmer Testing.

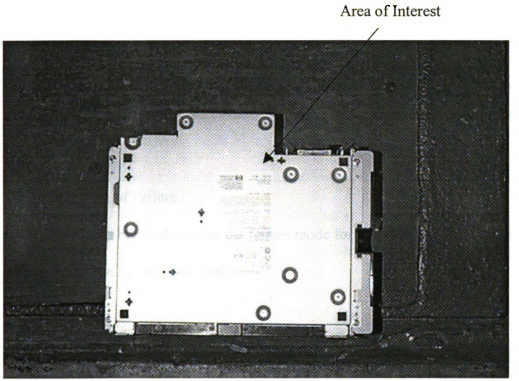

Figure 16. LaserJet Printer Sheet Metal.

waveform after excitation. Therefore, natural frequency was determined only by observing resonance on the vibration table during the sweep test. The reported natural frequencies in Table 3 are combined averages from the two methods since results were similar in each test.

Table 3. Natural Frequency of Test Models.

	PMMA	Metal Beam	119 Spring	Sheet Metal
Vibration Sweep	11	133	84	33

3.2.2 Definition of Failure

The table below describes the failure mode for each test model. The failure mode was chosen to be easily quantified.

Table 4. Failure Modes and Determination Method.

	Description of Failure	**Method of Failure Determination**
PMMA Beams	Complete break	Visual inspection
Metal Beams	Permanent deflection of 2.92 mm	Digital calipers
119 Spring	Permanent deflection of 1.63 mm	Digital calipers
Printer Sheet Metal	Permanent deflection of 3.50 mm	Digital calipers

3.2.3 Test Method for Finding Material Properties From Plastic Programmer Testing

The material properties A, B, x_0 and x_1 were found for all four components using the shock table plastic programmers, as outlined in Section 2.1. The test models were mounted on the shock table, and subjected to shocks on the plastic programmers at specified levels. Average N and ΔV were recorded from the drops and used for the data

points. Equations 6 through 8, 19 and 20 were then used to find the material properties. 30 specimens were tested at each of three different shock table heights for the PMMA and metal beams. Thirty 119 Springs were tested at each of two different heights. Five printers and eight printers, respectively, were tested at two different heights for the sheet metal properties. Fewer printers were tested due to the limited number of samples available.

3.2.4 Validation of Results From Plastic Programmer Testing

Freefall drops were used to validate the material properties found from the plastic programmer test method. Testing was done using the PMMA beams and sheet metal specimens. First, the values for x_0 and x_1 obtained from shock machine drops were inserted into the PROGRAM. The specimens were then subjected to freefall drops in cushioned boxes. Freefall drops were used to change the impact shock to the specimens as much as possible from the shock machine drops, thereby testing the model and program under very different conditions than were used to get material properties. Freefall drops were also chosen because they are simple and easy to perform. Dropping the packages repeatedly in flat-bottom drops until failure occurred (same failure as defined in the plastic programmer test), the number of drops to failure was recorded at each height. The observed number of drops to failure was then compared to the predicted number of drops from the PROGRAM and the digitized input shock obtained from the freefall drop.

For the PMMA beams, thirty separate beams mounted in the wood test fixture were tested using the free fall drop test machine from heights of 24, 26 and 30 inches. A

corrugated box was constructed to hold the wood test fixture and resilient foam (Dow Ethafoam^{Tm} 220, Figure 17). The input shock was captured by TP2 from an accelerometer mounted on the floor of the wood test fixture, and then translated into a .csv file format.

For the printer sheet metal, the unit was mounted onto a wood base, and placed into a box with foam for the freefall testing. Drops from 30 inches and 48 inches were done using the freefall drop tester. Because of limited number of units available, only three units were tested at 48 inches, and five units at 30 inches. More units were tested at the lower height since number of drops to failure varied more. Ethafoam 220^{Tm} was used as the cushion in the 30 inch drops, and a low density convoluted polyurethane cushion was used for the drops at 48 inches. These materials were chosen because of their availability and lower cost compared to neoprene. The accelerometer was mounted as shown in Figure 17, to capture the "input" pulse to the sheet metal. As mentioned before, accelerometer placement is very important since SRS assumes the pulse it is analyzing is an input to the component. Table 5 summarizes the test configurations used for the freefall drops.

Table 5. Freefall Drop Height Test Configurations.

	Freefall Drop Height	Cushion Thickness	Cushion Bearing Area
PMMA Beams	24 inches	2 inch	18 in^2
PMMA Beams	26 inches	1 inch	48 in^2
PMMA Beams	30 inches	1 inch	48 in^2
Sheet Metal	30 inches	1 inch	15 in^2
Sheet Metal	48 inches	1.75 inches	154 in^2

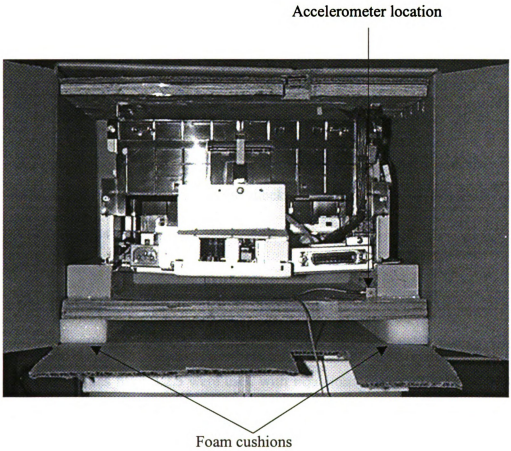

Figure 17. LaserJet Printer Mounted Onto Wood Base, Freefall Testing.

3.2.5 Test Method for Finding Material Properties From Freefall Drops

The material properties A, B, x_0 and x_1 were found for the PMMA beams and printer sheet metal using the freefall drop method outlined in Section 2.1 and the program in Appendix B. The procedure began by choosing the same freefall drop heights outlined in 3.2.4, and noting the number of drops to failure at each height. Representative analog shock pulses captured by accelerometer from each of the drop heights were digitized by TP2 and entered into the program using the .csv format. Using the observed number of drops to failure, starting values of x_0 and x_1 obtained from the plastic programmer method, and the .csv formatted input shock pulse, the program in Appendix B calculated the values for x_0 and x_1.

3.2.6 Validation of Results From Freefall Drops

To validate the material properties found from the freefall testing, the calculated x_0 and x_1 values from 3.2.5 were used with Equations 4 through 8 to calculate ΔV_{cr} and G_{cr} for the same number of drops to failure, N, as the plastic programmer test in 3.2.4. ΔV_{cr} and G_{cr} calculated from material properties found from freefall testing were then compared to ΔV_{cr} and G_{cr} calculated from material properties found from the plastic programmer testing.

3.2.7 Method for Generating Fatigue DBCs

Fatigue DBCs were constructed for the four products using ΔV_{cr} and G_{cr} calculated from the plastic programmer testing in 3.2.4. The fatigue DBC curves are

identical to traditional DBC curves, with the addition of number of drops to failure at a particular ΔV_{cr} and G_{cr}.

CHAPTER 4

RESULTS

4.1 RESULTS FOR MATERIAL PROPERTIES FROM PLASTIC PROGRAMMER TESTING

The material properties A, B, x_0 and x_1 for the four models were calculated from the plastic programmer testing method (3.2.3) and are summarized in Table 6. For the PMMA and metal beams, the higher B values in relation to A indicate more ductile materials, an expected result. Conversely, for the 119 Spring the smaller B value indicates a more brittle material, also as expected. The A and B values for the sheet metal indicate a somewhat ductile material. As described earlier, the value of A represents the smallest drop height in inches required for failure, and appears reasonable for each model. From Equation 10, adding the values of A and B gives the drop height for failure in one drop. The value for the PMMA beams appears somewhat high in a practical sense. However, the velocity change from the plastic programmer testing for each N was quite high to establish failure, and the equivalent freefall height causing damage in one drop is an extrapolation from this data.

The x_0 and x_1 values for the metal beams, springs and sheet metal in Table 6 all appear reasonable. Based on the size and shape of the actual models, the x_0 and x_1 values are plausible deflections. However x_0, and especially x_1, appear to be grossly overstated for the PMMA beams, given the fact the beams are less than seven inches in length to begin with. This raises the question whether the calculated material property values have real physical meaning, or are simply the result of a mathematical fit of a particular model to observed data. More importantly, if the material properties do not

Table 6. Summary of Material Properties From Plastic Programmer Testing.

		Plastic Programmer Data		Calculated Material Properties		
		Prescribed ΔV, in/sec Mean ± St. Dev.	Observed Drops to Failure, Mean ± St. Dev	Equivalent Drop Ht., in Equation 11 Mean ± St. Dev.	Fit Results, Equations 4 - 8	Corresponding Material Properties
PMMA Beams	1	144.9 ± 2.9	24.3 ± 11.9	27.2 ± .01	A = 22.8 B = 127.5 R^2 = .98	x_0 = 1.92 x_1 = 7.29
	2	169.5 ± 4.8	9.8 ± 4.5	37.2 ± .03		
	3	190.3 ± 6.6	5.2 ± 2.6	46.9 ± .06		
Metal Beams	1	126.3 ± 1.6	17.7 ± 4.0	20.7 ± 0	A = 17.2 B = 66.2 R^2 = 1.00	x_0 = 0.14 x_1 = 0.40
	2	137.7 ± 0.8	9.5 ± 1.1	24.5 ± 0		
	3	148.3 ± 1.1	5.8 ± 1.2	28.5 ± 0		
119 Spring	1	130.6 ± 1.1	5.3 ± 1.0	22.1 ± 0	A = 19.6 B = 13.1 R^2 = 1.0*	x_0 = 0.23 x_1 = 0.31
	2	141.9 ± 1.9	2.03 ± 0.2	26.1 ± .01		
		N/A	N/A	N/A		
Sheet Metal	1	124.8 ± 2.9	11.6 ± 5.8	20.1 ± .01	A = 17.8 B = 27.4 R^2 = 1.0*	x_0 = 0.57 x_1 = 1.00
	2	134.8 ± 1.8	4.8 ± 1.7	23.5 ± 0		
		N/A	N/A	N/A		

have real physical meaning, will the predictions of the PROGRAM using these material properties give inaccurate results when using them to predict number of drops to failure for different shock pulses. The x_0 and x_1 values are expected to be slightly higher in a dynamic test compared to a static test, and so performing a static deflection test will provide some basis for rationalizing the values found in Table 6. Several PMMA beams were tested in a static deflection test, and a representative curve is shown in Figure 18. The breakpoint x_1 reported is .55 inches, and x_0 can be estimated to be roughly .25 inches. These static values are about eight and fifteen times less than the dynamic x_0 and x_1 values in Table 6. Either the material properties for the PMMA beams were erroneously calculated from the plastic programmer testing, or the beams do not fit the idealized elastic, perfectly plastic model. To eliminate the possibility of calculation or data gathering error, the same material properties were found from freefall drop tests a few sections ahead in 4.3. Regardless of a lack of physical meaning, the calculated material properties were used with the elastic, perfectly plastic model to predict number of drops the failure for other types of input shocks by performing freefall drop testing as described in the next section.

4.2 RESULTS FOR VALIDATING MATERIAL PROPERTIES FOUND FROM PLASTIC PROGRAMMER TESTING

Shock pulses from freefall drop testing with the PMMA beams were entered into the PROGRAM to predict number of drops to failure using the material properties reported from plastic programmer testing in Table 6. The predicted number of drops to failure from the PROGRAM were compared to the observed number of drops to failure,

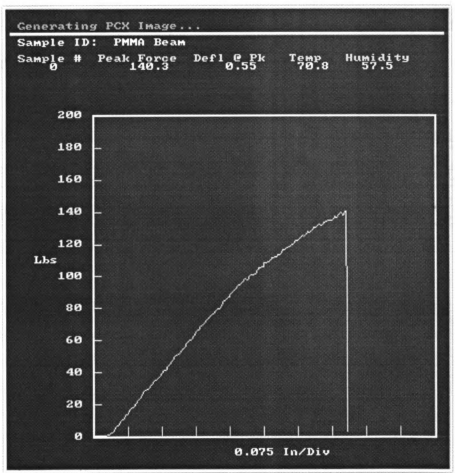

Figure 18. Static Force Versus Deflection Curve For PMMA Beam.

and the results are summarized in Table 7. Representative shock pulses were entered into the PROGRAM from the 30, 26 and 24 inch drop heights. 30 separate beams were tested at each drop height.

Similarly for the sheet metal, representative shock pulses from the 30 and 48 inch drop heights were entered into the PROGRAM, along with the material properties from Table 6, to predict number of drops to failure. Actual number of drops to failure was compared with the predicted number, and is summarized in Table 8.

Table 7. Actual and Predicted Drops to Failure, Freefall Testing, PMMA Beam.

	30 inches		26 inches		24 inches	
Method	Avg.	St. Dev.	Avg.	St. Dev.	Avg.	St. Dev.
Actual	3.8	2.1	5.2	2.9	7.9	5.3
Predicted	3.9	1.1	4.3	0.3	8.7	2.2

Table 8. Actual and Predicted Drops to Failure, Freefall Testing, Sheet Metal.

	30 inches		48 inches	
Method	Average	St. Dev.	Average	St. Dev.
Actual	9.2	3.6	5.3	1.2
Predicted	1.8	.5	1.2	.1

The results show good agreement for the PMMA beams. The actual and predicted number of drops to failure was always within one drop for all drop heights. This suggests the material property values, though maybe not physically descriptive, are useful for predicting damage from different shock pulses. The standard deviation from the predicted number of drops is about half from the actual observed, a reflection of the material variability since the PROGRAM's only variation is the shock input itself.

The results for the printer sheet metal show the predicted number of drops to be too low compared to the actual. The large standard deviation in the observed number of drops compared to the predicted indicates material variability plays a significant role in these results. Material variability is particularly important since relatively few samples were tested at each drop height, both for the plastic programmer testing and the freefall testing. This is a practical limitation since it was prohibitively expensive to test many products. Trade-offs may need to be made between cost and accurate data. It is also possible the sheet metal is not a single degree of freedom spring/mass system, an assumption of Equation 24. The sheet metal has a PCB board with many electronic components affixed to it, making it a complex arrangement of many spring/mass systems. The model in this study may not accurately represent this type of system. The location of the accelerometer to capture the input shock may also have affected predicted results. As shown in Figure 17, the accelerometer was mounted to the wood test fixture, which itself is a spring/mass system, able to flex during a freefall drop. The true input shock to the sheet metal may in fact have been something different than the shock captured by the accelerometer mounted to the base of the wood test fixture.

4.3 RESULTS FOR MATERIAL PROPERTIES FOUND IN FREEFALL DROPS

Using the procedure outlined in 3.2.5, material properties A, B, x_0 and x_1 for the PMMA beams and the printer sheet metal were again found from the freefall drop testing, and are summarized in Table 9. The results for x_0 and x_1 show good agreement with those in Table 6 obtained from plastic programmer testing. The agreement is within 3% for the PMMA beams, and within 12% for the printer sheet metal. As noted in the

previous section, it is possible the accelerometer location on the wood test fixture did not capture the true input shock to the sheet metal, and may explain the difference in the calculated material property values. This points out one benefit for using the plastic programmers; the rigid, heavy shock table eliminates the flexing of the wood test fixture during the drop event, and may provide a more consistent and true input shock.

For the PMMA beams, the good agreement for x_0 and x_1 between the freefall method and the plastic programmer method eliminates the possibility of calculation or data gathering error for the material properties. This suggests that the model does not accurately describe the PMMA beams. Despite the model's inability to predict physically meaningful material properties for the PMMA beams, it still fits values that are able to correctly predict number of drops to failure from very different shock pulses when used in the fatigue shock response PROGRAM. This is an acceptable situation, since the goal is to predict number of drops to failure, not isolate actual material properties, which have little independent value.

Table 9. Summary of Calculated Material Properties From Freefall Testing.

	Drop Height, inches	Freefall Drop Test Data	Calculated Material Properties	
		Observed Drops to Failure, Mean ± St. Dev.	Fit Results, Equations 4 - 8	Material Properties, inches
PMMA Beams	24	7.9 ± 5.3	A = 24.2 B = 130.0 R^2 = 0.99	x_0 = 1.97 x_1 = 7.30
	26	5.2 ± 2.9		
	30	3.8 ± 2.1		
Sheet Metal	30	9.2 ± 3.6	A = 23.5 B = 32.5 R^2 = 1.0*	x_0 = 0.65 x_1 = 1.10
	48	5.3 ± 1.2		

*Only two data points gives perfect fit for straight line

4.4 RESULTS FOR VALIDATING MATERIAL PROPERTIES FOUND FROM FREEFALL DROP TESTING

Table 10 summarizes the comparison between the calculated ΔV_{cr} and G_{cr} (Equations 4 and 5) using x_0 and x_1 values found from freefall testing and those found in the plastic programmer testing, both at the same number of drops to failure, N. Results show less than 3% difference between the two methods for the PMMA beams, and about 13% for the sheet metal.

Table 10. Comparison of Calculated ΔVcr and Gcr.

		PMMA Beams			Sheet Metal	
	N	Plastic Programmers	FreeFall	N	Plastic Programmers	FreeFall
ΔVcr, in:	24.3	147.2	151.1	11.5	124.8	144.6
Gcr:		13.1	13.5		33.4	38.7
ΔVcr, in:	9.8	166.2	170.1	4.8	134.8	151.3
Gcr:		14.5	14.9		35.8	40.3
ΔVcr, in:	5.2	191.5	195.3	N/A	N/A	N/A
Gcr:		16.0	16.4		N/A	N/A

4.5 RESULTS FOR GENERATING FATIGUE DAMAGE BOUNDARY CURVES

To generate fatigue DBCs, only x_0 and x_1 are needed, either from the plastic programmer or freefall test method. ΔV_{cr} and G_{cr} are then calculated from Equations 4 through 7 for any N. To demonstrate this, Table 11 was constructed using x_0 and x_1 from the plastic programmer testing method and three different values of N. This information is plotted in Figures 19 through 22. The rounded knee has been squared off for simplicity, and the axis formatted to make the curves easy to read on the pages.

Table 11. Data For Constructing Fatigue DBCs.

	N = 1		N = 5		N = 10	
	ΔVcr, in	Gcr	ΔVcr, in	Gcr	ΔVcr, in	Gcr
PMMA Beam	340.8	20.6	193.2	16.1	165.8	14.5
Metal Beam	253.9	206.7	153.4	159.3	135.7	144.8
119 Spring	158.9	105.1	131.0	89.3	127.1	86.8
Sheet Metal	186.9	45.2	134.1	35.7	126.0	33.7

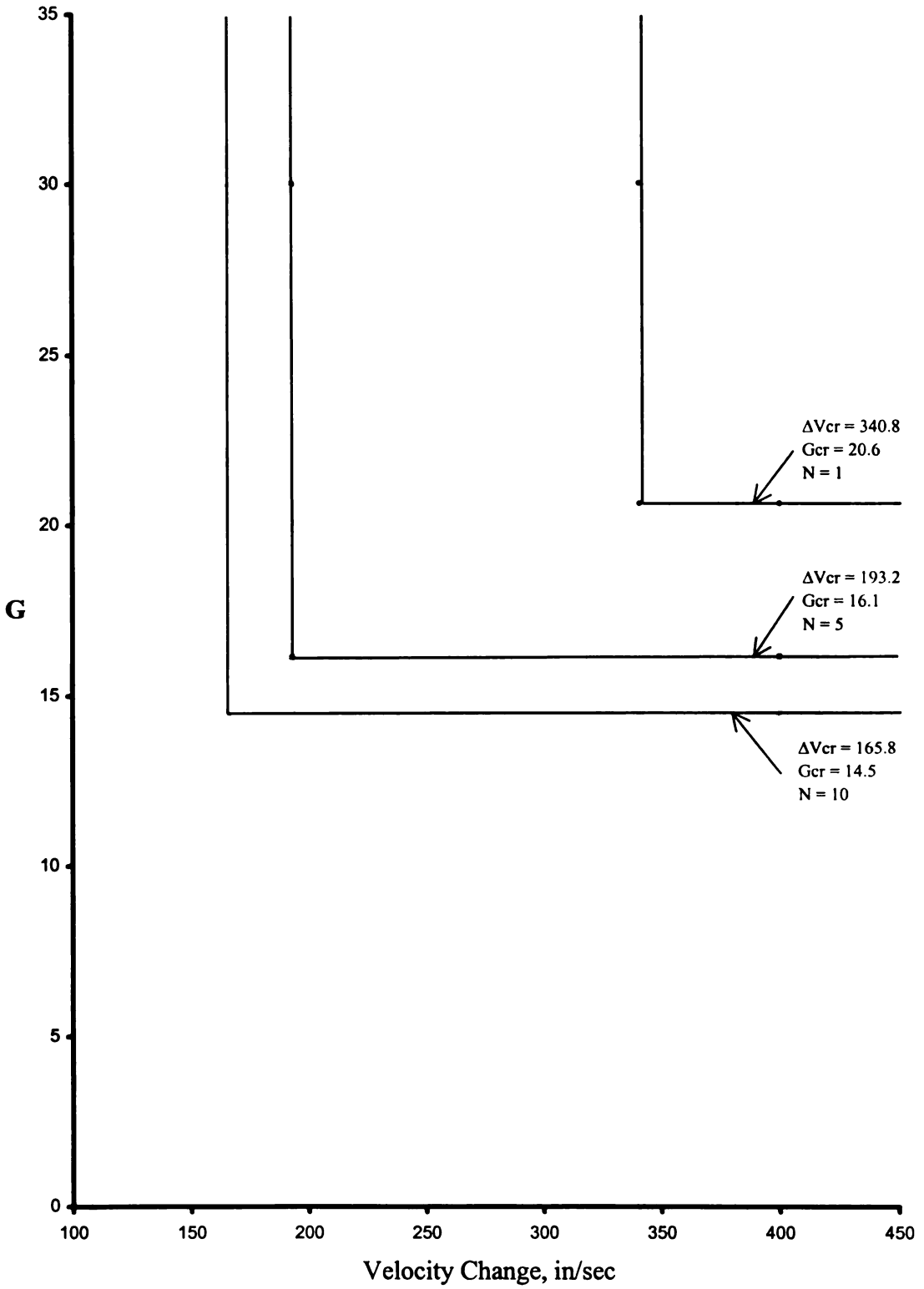

Figure 19. DBCs for PMMA Beams.

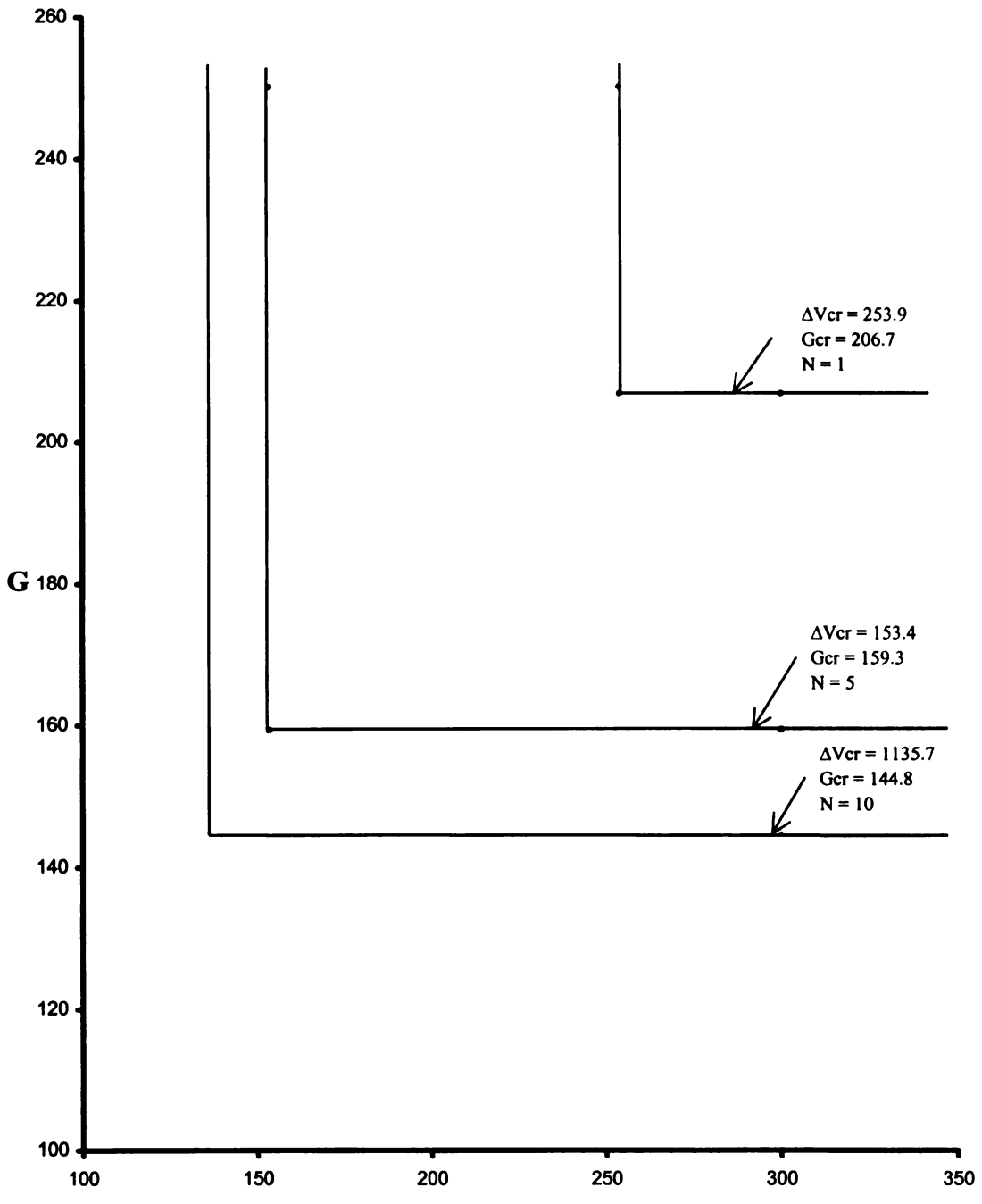

Figure 20. DBCs for Metal Beams.

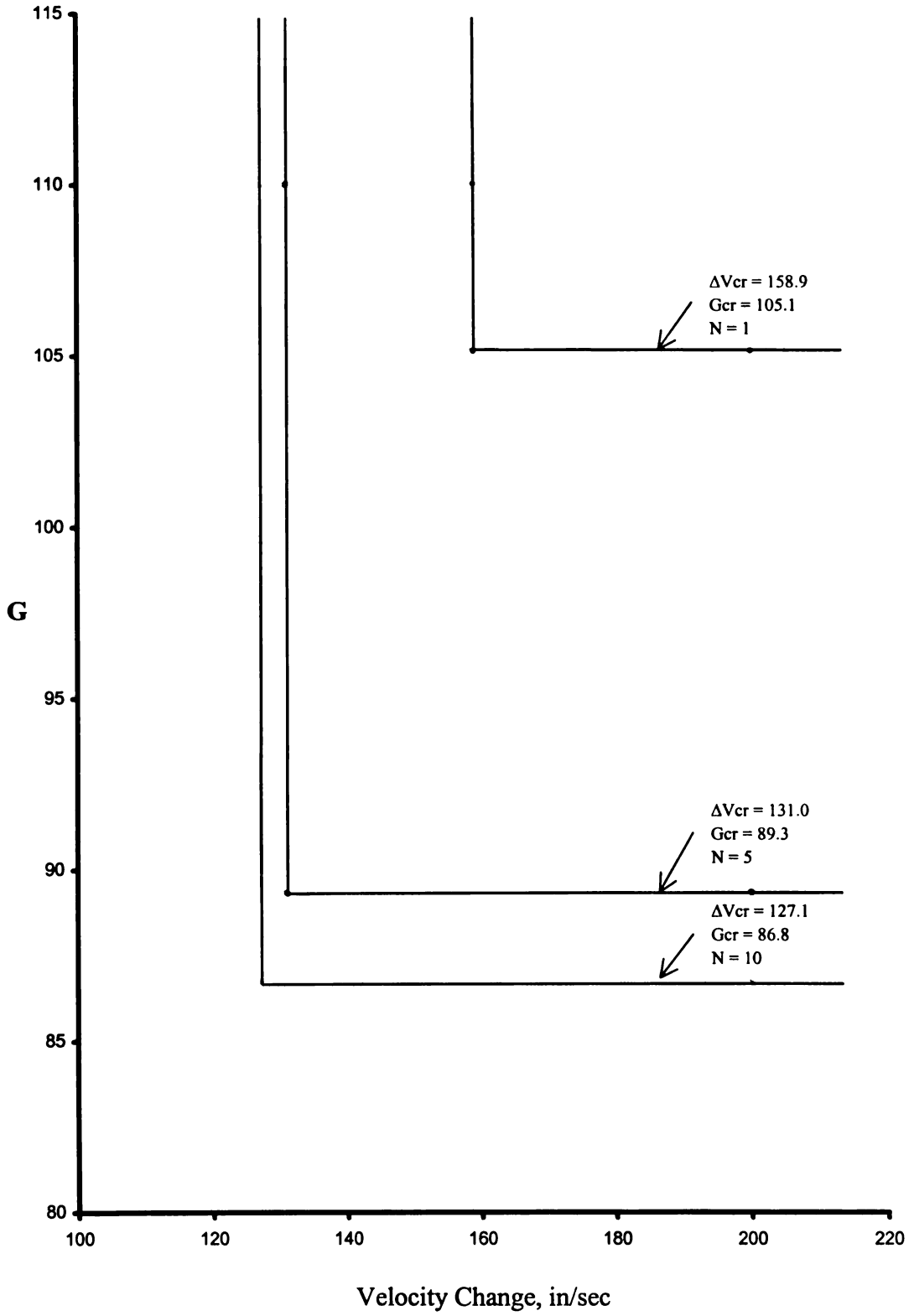

Figure 21. DBCs for 119 Spring.

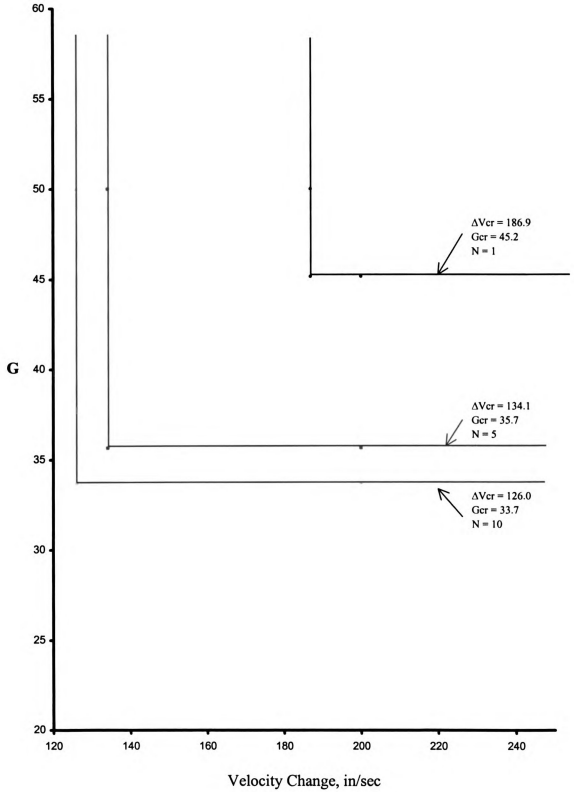

Figure 22. DBCs for Sheet Metal.

4.6 DISCUSSION OF ERRORS AND IMPACT ON RESULTS

Observing Tables 7 and 8 points out some discrepancy between the actual and predicted number of drops to failure. There are several possible explanations for these discrepancies. First and foremost is material variability. Number of drops to failure from the plastic programmer or freefall test (to determine A and B) will affect x_0 and x_1. One approach to handle this variability is to vary N within experimentally observed ranges and generate the corresponding x_0 and x_1 values. The PROGRAM can then use these values to predict a range for number of drops to failure.

Another approach is to directly observe the variation in x_0 and x_1 by conducting a static compression test. As pointed out earlier, this is not a recommended method for determining the material properties. Nevertheless, this test may prove useful for establishing upper and lower limits on x_0 and x_1. Testing of this sort was done using the PMMA beams (Figure 18), and the average and standard deviation for x_0 and x_1 were established. The percent standard deviation was then used with the average x_0 and x_1 values calculated from Equations 19 and 20 to describe ranges for x_0 and x_1 values. Table 12 shows the results for the PMMA beams, σ representing standard deviation. The results in Table 7 suggest no adjustment is necessary in x_0 and x_1 to correctly predict average number of drops to failure for the PMMA beams. However, the smaller standard deviation from the predicted number of drops suggests the PROGRAM might not account for variations in number of drops to failure when working with smaller sample sizes. Conversely, varying x_0 and x_1 might be beneficial for the sheet metal, based on the results in Table 8, and a table could be constructed for the sheet metal similar to Table 12.

Table 12. Variation of x_0 and x_1 From Static Deflection Test.

	x_0, in	x_1, in
Average	1.92	7.29
1 σ	1.75 – 2.09	5.85 - 8.73
2 σ	1.57 – 2.27	4.42 - 10.16
3 σ	1.40 – 2.44	2.99 - 11.59

Another potential cause of error is the choice of N during the procedure to find A and B. As previously discussed, the true value of N occurs somewhere between the drop causing observed failure and the previous drop. Table 13 shows how N from plastic programmer testing would vary depending on how N is chosen. "High" refers to N when damage is observed, "Low" for the drop before observed failure, and "Average" for the average of the "High" and "Low" N values. x_0 and x_1 are subsequently affected by the choice of N as shown in Table 14. Eventually number of drops to failure predicted by the PROGRAM would be affected. Similar tables could be constructed for the freefall test method showing how x_0 and x_1 are affected by the choice of N.

Table 13. Values For Number of Drops to Failure.

		Number of Drops to Failure, High	**Number of Drops to Failure, Average**	**Number of Drops to Failure, Low**
PMMA Beam	1	24.27	23.77	23.27
	2	9.83	9.33	8.83
	3	5.17	4.67	4.17
Metal Beam	1	17.73	17.23	16.73
	2	9.53	9.03	8.53
	3	5.77	5.27	4.77
119 Spring	1	5.30	4.80	4.30
	2	2.03	1.53	1.03
Sheet Metal	1	11.60	11.10	10.60
	2	4.78	4.28	3.78

Table 14. Effect of N on Material Properties x_0 and x_1.

		PMMA		Metal		119 Spring		Sheet Metal	
	Method	x_0	x_1	x_0	x_1	x_0	x_1	x_0	x_1
N High	PP	1.92	7.29	0.14	0.40	0.23	0.31	0.57	1.00
N Average	PP	1.95	6.60	0.14	0.37	0.24	0.29	0.57	0.94
N Low	PP	1.98	5.95	0.14	0.34	0.24	0.27	0.57	0.88
N High	FF	1.97	7.30	N/A	N/A	N/A	N/A	0.65	1.10
N Average	FF	1.93	7.26	N/A	N/A	N/A	N/A	0.65	1.08
N Low	FF	1.89	7.22	N/A	N/A	N/A	N/A	0.64	1.13

A third but less important influence on x_0 and x_1 is the variation of ΔV during the determination of A and B. The ΔV from the plastic programmers can usually be held to within +/- 5%, which should minimize its effect on determining A and B. This is demonstrated in Table 5, where the R^2 value (curve fit) is very high, indicating drop height (ΔV) does not vary in a way that strongly influences the calculation of x_0 and x_1 from A and B. However, the ΔV in freefall drops may be more difficult to control, since factors such as cushion and corrugated degeneration can occur over multiple drops. Since the package falls without restraint, it is also more difficult to precisely control drop orientations, which may lead to small changes in the shock pulse shape, as discussed in 2.1.3. Thus, one might expect less accuracy for the material properties generated from freefall testing.

Another possible influence on results is natural frequency determination. Resonance usually occurs within a range of frequencies, which is in contrast to the single number used by the PROGRAM for shock response analysis.

Other errors introduced include such things as accelerometer calibration, electromagnetic noise captured by accelerometer cables, and computer round off error. Care was taken to prove the accelerometers were calibrated properly, and normal precautions were taken during data gathering so that these errors should be regarded as incidental.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

5.1 CONCLUSIONS

Material properties are easily obtained from the plastic programmer test method because the velocity change and number of drops to failure are used directly in a simple regression scheme. This is made possible because the shock generated by the plastic programmers is a "spike", for which the response of the component has a single closed form solution. The material properties found from the freefall test method require a computer program because the shock is no longer a spike. Still, they showed good agreement with those obtained in the plastic programmer test method. Thus, even with shock pulse and cushion variability, the freefall method appears to be sufficient for obtaining the characteristic material properties needed for the model. The procedure outlined in Section 2.1.3 and Appendix B provide a straightforward and easy method for determining material properties without the need for a shock table.

For both the PMMA beams and printer sheet metal, the freefall method predicts material properties similar to those obtained from the plastic programmer testing, but the plastic programmer method is recommended for obtaining them. The reason is two-fold. First, the plastic programmer method eliminates potential complications when deciding accelerometer location, and secondly there is no concern with cushion material performance from drop to drop. The shock response PROGRAM in Appendix C can then be used with these material properties to predict number of drops to failure for any

input shock. Fatigue DBCs can also be constructed using Equations 4 through 7. All of this can be done without an expensive shock table.

As demonstrated with the PMMA beams, the SRS PROGRAM utilizing the elastic, perfectly plastic model and dynamic material properties accurately predicts number of drops to failure for input shocks which are very different from the ones used to find the material properties, even when the material properties appear to have no physical meaning. Thus, the model appears to correctly extrapolate material property values that describe behavior during freefalls from plastic programmer data. For the sheet metal, using material properties obtained from plastic programmer testing with the PROGRAM predicts responses in freefall drops to be more severe than actual, most likely because the sheet metal is not a single degree of freedom system. It is also possible flexing of the wood test fixture during freefall drops complicates the input shock to the component, adversely affecting the predictions of the SRS PROGRAM.

There are several sources of experimental error that affect the calculations of material properties, and subsequently the predictions made by the SRS PROGRAM. Small changes in velocity change from drop to drop have some effect on calculating material properties. Choosing the N corresponding to failure also affects calculated material properties. The largest contributor to experimental error is material variability, evidenced by the large variations in the number of drops to failure during plastic programmer and freefall testing. These experimental errors are unavoidable, and need consideration during fatigue fragility testing.

5.2 FUTURE WORK

The most beneficial outcome from this new fragility model is that once material properties are defined, they can be used to predict percent damage caused by any type of input shock, including dissimilar multiple shocks to the same component. Shock pulses from different drop heights or onto different surfaces could be analyzed for their cumulative damage effect on the component. This is done by simply entering each shock pulse into the PROGRAM to predict percent damage, and adding the results. Future work could concentrate on verifying this use by subjecting a test specimen to several different shock pulses, and comparing the predicted drops to failure with actual number of drops to failure.

Other areas for future work include the following:

•SRS is sensitive to the input shock used for analysis. More work should be done to analyze the effects of SRS predictions from differing accelerometer locations and mountings.

•The elastic/perfectly plastic model assumes natural frequency of the critical component remains the same from drop to drop – the stretching and releasing of the spring does not alter its natural frequency. Checking the component's natural frequency after each drop could serve as an additional validation for this assumption.

•This study focused on the ability of the PROGRAM to predict only the total number of drops to failure given identical multiple shocks. The permanent deformation from each drop should be checked to see if it is constant as the model predicts.

•Much has been said about obtaining the material properties x_0 and x_1. Another possible method for obtaining them is to use high-speed video during dynamic testing. A method similar to the ones described in earlier works [20] might prove useful.

•As noted in the Introduction, it was observed from earlier work that the fatigue model might satisfactorily describe crack propagation failure, and more investigation of this may be warranted.

•The fatigue shock response PROGRAM in Appendix C can easily be modified to calculate shock response spectra over any desired range of frequencies. The program for finding material properties in freefall testing outlined in Appendix B could be modified to include a print-out of the calculated DBCs.

•Investigation of weighting methods mentioned in Section 2.1 may prove useful for particular materials.

•Since S-N curve data already exists for many materials, it may prove worthwhile to investigate developing a method for combining the elastic, perfectly plastic model in this study with the existing S-N data. This may reduce or eliminate lab testing for generating material properties.

•And finally, useful testing can be done on other models of interest to validate the elastic, perfectly plastic model used with SRS.

APPENDIX A

APPENDIX A

RECOVERING OTHER MODELS

The rigid, perfectly plastic model [7] can be recovered from the elastic, perfectly plastic model by first incorporating Equation 8 into Equations 6 and 7 to get A and B in terms of x_0 before the limit is taken:

$$A = \frac{F_0 g}{x_0 W} \cdot \frac{x_0^2}{2g} = \frac{F_0 x_0}{2W} \tag{35}$$

$$B = \frac{F_0 g}{x_0 W} \cdot \frac{x_0^2}{g}\left(\frac{x_1}{x_0} - 1\right) = \frac{F_0}{W}\left(x_1 - x_0\right) \tag{36}$$

Substituting into Equation 5 and simplifying yields

$$G_{cr} = \frac{F_0}{W} \cdot \frac{Nx_0 + 2x_1 - 2x_0}{2Nx_0 + 2x_1 - 2x_0} \tag{37}$$

Taking the limit of x_0 as it approaches zero gives:

$$\lim_{x_0 \to 0} G_{cr} = \frac{F_0}{W} = \frac{A_{cr}}{g} \tag{38}$$

which corresponds with the result in [7]. Similarly for ΔV_{cr}, substituting into Equation 4 and simplifying yields

$$\lim_{x_0 \to 0} \Delta V_{cr} = \sqrt{2g \cdot \frac{F_0 x_1}{WN}} = \sqrt{2\left(\frac{F_0}{m}\right) \cdot \frac{x_1}{N}} \tag{39}$$

which is the same expressions found in [7].

The brittle model, where $x_1 = x_0$, can also be recovered from the elastic, perfectly plastic model by observing that B = 0 in Equation 7, so that Equations 4 and 5 read

$$\Delta V_{cr} = \sqrt{2g \frac{\omega^2 x_0^2}{2g}} = 2\pi f_n x_0 \tag{40}$$

$$G_{cr} = \omega \sqrt{\frac{2\dfrac{\omega^2 x_0^2}{2g}}{g}} \left(\frac{\dfrac{\omega^2 x_0^2}{2g}}{2\dfrac{\omega^2 x_0^2}{2g}} \right) = \frac{(2\pi f_n)^2 x_0}{2g} \tag{41}$$

APPENDIX B

APPENDIX B

MATERIAL PROPERTIES FROM FREEFALL TESTING

The following is a summary outlining the steps for find material properties x_0 and x_1 from freefall drop testing.

1. Construct a test package for the test specimen. Mount an accelerometer in a location to capture the input shock to the specimen.
2. Drop the test package from a known height, recording the shock pulse from the accelerometer.
3. Repeat dropping from the same height until failure occurs, recording number of drops to failure.
4. Repeat Steps 2 and 3 for a different drop height.
5. Calculate starting values for x_0 and x_1 using the same regression equations (Equations 16 through 18) as for the plastic programmer method.
6. Using the material properties from Step 5 above, and the Program in Appendix C, calculate the predicted number of drops to failure for each drop height in Steps 2 through 4.
7. Calculate SSE for actual and predicted number of drops to failure.
8. Systematically vary x_0 and x_1 and repeat Steps 6 and 7 until the SSE is minimized.

APPENDIX C

APPENDIX C

SRS PROGRAM CODE

The following is Visual Basic 5.0 code for the SRS algorithm incorporating the elastic, perfectly plastic stress-strain curve to accommodate fatigue failure. The code is written to demonstrate the algorithm described in Chapter 2, not to necessarily represent a commercial software application.

```
'=============================================================

'Modified: 16 December 1998
'Main engine for algorithm:
Option Explicit
Dim lcv, intNumRec, intCounterForArray As Integer   'counter variables
Dim sngSampleFn, sngX0, sngX1, sngDuration, sngDT, sngZ3, sngFn As Single
Dim sngPRIG, sngZMax, sngKM, sngZ, sngZDot, sngZ2, sngF0 As Single
Dim sngT, sngOA, sngZDDot, sngIA, sngFZ2, sngPercent, sngDrops As Single
Dim strFileName As String
Dim ShockData As ShockData  'udt for input shock
Dim strTemp As String
Dim strTemp2(2000) As String
' Dim sngDR, sngCM
'=============================================================
```

```
Private Sub Form_Load()
frmMain.Top = (Screen.Height - frmMain.Height) / 2  'center form in screen of any size
   frmMain.Left = (Screen.Width - frmMain.Width) / 2
   fraResults.Visible = False
   frmMain.Show    'form must be loaded or shown before SetFocus will work
   txtX0.SetFocus  'must show form before any objects in the form can have SetFocus

   With graShockData
      .Visible = False
      .DataGrid.RowCount = 5000
      .ColumnCount = 4
   End With

   With graForceDeflection     'set up graph for force/deflection curve
      .Visible = False
      .DataGrid.RowCount = 5000
      .DataGrid.ColumnCount = 2
      .Plot.Axis(VtChAxisIdX).AxisTitle = "Deflection (in)"
      .Plot.Axis(VtChAxisIdY).AxisTitle = "Spring Force (lbs)"
   End With
End Sub
'=============================================================================
```

```
Private Sub Form_Unload(Cancel As Integer)
Const conBtns = vbYesNoCancel + vbExclamation + vbDefaultButton3 +
        vbApplicationModal
   'Const conmsg = "Do you want to save the current information?"
   Dim intUserResponse As Integer
   'If blnChange = True Then
   intUserResponse = MsgBox("Are you sure you want to exit?", conBtns, "SRS")
      If intUserResponse = vbYes Then
         End
      Else
         Cancel = 1
      End If
   'add dialog to ask if want to save?
End Sub
'=============================================================================
Private Sub mnuAboutAbout_Click()
   frmSplash.Show
End Sub
'=============================================================================
Private Sub mnuCalculateSR_Click()
frmMain.MousePointer = vbHourglass  'mouse pointer turns to hourglass

'Open "e:\def_data" For Output As #2     'check values calculated by program
```

```
'reset variables to 0, allowing repeated opening of files and calc's w/o quitting application
    sngZ = 0
    lcv = 0
    sngDuration = 0
    sngDT = 0
    sngZ3 = 0
    sngPRIG = 0
    sngZMax = 0
    sngKM = 0
    sngZ = 0
    sngZDot = 0
    sngZ2 = 0
    sngF0 = 0
    sngT = 0
    sngOA = 0
    sngZDDot = 0
    sngIA = 0
    sngFZ2 = 0
    sngPercent = 0
    sngDrops = 0
    intCounterForArray = 0

    For lcv = 1 To 5000        'clear out arrays for force vs deflection graph
```

```
    graForceDeflection.DataGrid.SetData lcv, 1, 0, 1     'row, column, value, not visible
    graForceDeflection.DataGrid.SetData lcv, 2, 0, 1
Next lcv

'read values from form into variables
sngSampleFn = Val(txtSampleFn.Text)
'sngDR = Val(txtDampRatio.Text)
sngX0 = Val(txtX0.Text)
sngX1 = Val(txtX1.Text)
sngFn = Val(txtFn.Text)

sngDuration = ShockData.sngMS(intNumRec)    'duration will be equal to last ms
        reading from the inported pulse
sngDT = 1 / sngSampleFn  'set time step based on sampling frequency supplied by user
sngKM = (2 * 3.14159 * sngFn) ^ 2
For sngT = sngDT To sngDuration Step sngDT      'run program to end of input pulse
    intCounterForArray = intCounterForArray + 1     'keep track of iterations

    'interpolate to get input G level from shock pulse
    For lcv = 1 To intNumRec - 1
        If sngT >= ShockData.sngMS(lcv) And sngT <= ShockData.sngMS(lcv + 1) Then
            sngIA = ShockData.sngG(lcv) + ((ShockData.sngG(lcv + 1)-
                ShockData.sngG(lcv)) * (sngT - ShockData.sngMS(lcv)) _
```

```
        / (ShockData.sngMS(lcv + 1) - ShockData.sngMS(lcv)))
      End If
    Next lcv

    'place value into datagrid to be plotted
    graForceDeflection.DataGrid.SetData intCounterForArray, 1, sngZ, 0
    graForceDeflection.DataGrid.SetData intCounterForArray, 2, sngF0, 0

    sngZ = sngZ + sngZDot * sngDT + sngZDDot * sngDT ^ 2 / 2    'set value for
      position

    'Write #2, "sngz= " & sngZ & "= " & sngF0

    If sngZ <= -sngX0 Then Exit For    'exit loop if in elastic region, at elastic limit
      deflection and negative

If Abs(sngZ) > sngZMax Then sngZMax = sngZ      'keep track of maximum position

    'check for position exceeding x1
    If Abs(sngZMax) > sngX1 Then
      lblCalcDrops.Caption = "This shock breaks the element"
      lblCalcDamage.Caption = "100%"
      Exit For
```

```
End If


'Elastic Loading Criteria
   If (Abs(sngZ) <= sngX0 And sngFZ2 = 0) Then     'use sngFZ2 as flag; 0
            indicates not in unloading
      sngZDot = sngZDot + sngZDDot * sngDT
      sngZDDot = sngIA - (sngKM * sngZ) '- (sngCM * sngZDot)
      sngF0 = sngKM * sngZ
      'Write #2, "Elastic." & "F= " & sngF0 & "z= " & sngZ & "zdot= " & sngZDot
   End If


'Plastic Loading Criteria
   If (sngZ > sngX0 And sngZDot > 0) Then
      sngZDot = sngZDot + sngZDDot * sngDT
      sngZDDot = sngIA - (sngKM * sngX0) '- (sngCM * sngZDot)
      sngF0 = sngKM * sngX0
      sngFZ2 = 0.1   'flag to keep program from going back into elastic routine
      'Write #2, "Plastic Loading." & "F= " & sngF0 & "z= " & sngZ & "zdot= " &
            sngZDot
   End If


'Plastic Unloading Criteria
      'sngFZ2 is flag to keep from elastic routine
```

```
    If (sngZDot < 0 And Abs(sngF0) <= (sngKM * sngX0)) And sngFZ2 <> 0 Then
      sngZDot = sngZDot + sngZDDot * sngDT
      sngZDDot = sngIA - (sngKM * (sngX0 - sngZMax + sngZ)) '- (sngCM *
          sngZDot)
      sngF0 = sngKM * (sngX0 - (sngZMax - sngZ))
      sngFZ2 = sngF0  'set sngFZ2, and hold for plastic loading
      'Write #2, "Plastic Unloading." & "F= " & sngF0 & "z= " & sngZ & "zdot= " &
          sngZDot
      If (sngZDot >= 0 And Abs(sngF0) <= (sngKM * sngX0)) Then Exit For
    End If

'Plastic Loading, Tension Criteria
    If (sngZDot < 0 And Abs(sngF0) >= (sngKM * sngX0)) Then
      sngZDot = sngZDot - sngZDDot * sngDT
      sngZDDot = sngIA - (sngKM * sngZ2) '+ (sngCM * sngZDot)
      If sngZ < sngZ2 Then sngZ3 = sngZ
      sngF0 = sngFZ2      'get sngFZ2 from end of plastic unloading; should be
          (sngkm*sngx0)
      'Write #2, "Plastic Loading, tension." & "F= " & sngF0 & "z= " & sngZ &
          "zdot= " & sngZDot

      'exit loop if velocity changes to positive, or when position equals sngx0 – this
          forces  conformity to Bauschinger model
```

```
            If sngZDot >= 0 Or Abs(sngZ) >= sngX0 Then Exit For
        End If
'calculate output acceleration, place result into datagrid
    sngOA = sngKM * sngZ '+ sngCM * sngZDot
    graShockData.DataGrid.SetData intCounterForArray, 3, sngT, 0
    graShockData.DataGrid.SetData intCounterForArray, 4, sngOA / 386.4, 0

    'keep track of peak output G
    If Abs(sngOA) > Abs(sngPRIG) Then sngPRIG = sngOA
  Next sngT

'=============================================================

'check for not reaching elastic limit
If Abs(sngZMax) < sngX0 Then
  lblCalcDamage.Caption = "No Damage"
  lblCalcDrops.Caption = "Below Elastic Limit"
End If

'calculate percent damage, etc.
If Abs(sngZMax) > sngX0 And Abs(sngZMax) < sngX1 Then
  sngPercent = Abs((Abs(sngZMax) - sngX0) / (sngX1 - sngX0))
  sngDrops = Int(1 / sngPercent)
  lblCalcDamage.Caption = Format(sngPercent, "percent")
```

```
    lblCalcDrops.Caption = Format(sngDrops, "##.#0")
End If

lblCalcPeakG.Caption = Format(sngPRIG / 386.4, "####.#0")
lblCalcDisplacement.Caption = Format(sngZMax, "##.#0")
fraResults.Visible = True

graShockData.Plot.UniformAxis = False
graShockData.Title.Text = "Shock Input and Shock Response"
graForceDeflection.Plot.UniformAxis = False
graForceDeflection.Visible = True
frmMain.MousePointer = vbDefault    'return mouse to regular pointer
'Close #2
End Sub
'=============================================================

Private Sub mnuFileExit_Click()
    Unload Me
End Sub
'=============================================================

Private Sub mnuFilePrintAll_Click()
    PrintForm
End Sub
'=============================================================
```

```
Private Sub mnuFilePrintData_Click()
   PrintForm
End Sub
'=============================================================

Private Sub mnuFileRetrieveSP_Click()
   graShockData.Visible = False
   frmMain.MousePointer = vbHourglass

   With dlgCommon
      .DialogTitle = "Retrieve Input Shock Pulse"
      .Flags = cdlOFNHideReadOnly + cdlOFNFileMustExist 'hide read only box
      .Filter = "Text Files(*.txt)|*.txt|CSV Files(*.csv)|*.csv"
      .filename = ""
      .ShowOpen
   End With

   'if cancel is selected, exit sub without loading shock pulse data
   On Error Resume Next
   If Err.Number = 75 Then Exit Sub

   'clear out the array from any previous shock pulse data
   For lcv = 0 To 2000
      strTemp2(lcv) = 0
```

```
    ShockData.sngMS(lcv) = 0
    ShockData.sngG(lcv) = 0
Next lcv

For lcv = 1 To 5000
    graShockData.DataGrid.SetData lcv, 1, 0, 1
    graShockData.DataGrid.SetData lcv, 2, 0, 1
    graShockData.DataGrid.SetData lcv, 3, 0, 1
    graShockData.DataGrid.SetData lcv, 4, 0, 1
Next lcv

Open dlgCommon.filename For Input As #1
intNumRec = 0

If dlgCommon.filename <> "" And UCase(Right(dlgCommon.filename, 3)) = "TXT"
        Then
'input text file, formatted with MS and G, seperated by commas
    Do While Not EOF(1)
        intNumRec = intNumRec + 1
        Input #1, ShockData.sngMS(intNumRec), ShockData.sngG(intNumRec)
        ShockData.sngMS(intNumRec) = ShockData.sngMS(intNumRec) / 1000

        'set graph array data to time (sec) and G data
```

```
        With graShockData.DataGrid
          .SetData intNumRec, 1, ShockData.sngMS(intNumRec), 0
          .SetData intNumRec, 2, ShockData.sngG(intNumRec), 0
          .SetData intNumRec, 3, 0, 1
          .SetData intNumRec, 4, 0, 1
        End With

        ShockData.sngG(intNumRec) = ShockData.sngG(intNumRec) * 386.4
      Loop
    Close #1

    frmMain.Caption = dlgCommon.filename
  End If
'open CSV file, directly from TP2
  If dlgCommon.filename <> "" And UCase(Right(dlgCommon.filename, 3)) = "CSV"
      Then
    Line Input #1, strTemp  'get rid of first line
      Do While Not EOF(1)
        intNumRec = intNumRec + 1   'update record counter
        Input #1, ShockData.sngMS(intNumRec), ShockData.sngG(intNumRec),
              strTemp2(intNumRec)
        ShockData.sngMS(intNumRec) = ShockData.sngMS(intNumRec) / 1000
```

```
        'fill graph array with time and G data
        With graShockData.DataGrid
          .SetData intNumRec, 1, ShockData.sngMS(intNumRec), 0
          .SetData intNumRec, 2, ShockData.sngG(intNumRec), 0
          .SetData intNumRec, 3, 0, 1
          .SetData intNumRec, 4, 0, 1
        End With

        ShockData.sngG(intNumRec) = ShockData.sngG(intNumRec) * 386.4
      Loop

    Close #1
    frmMain.Caption = dlgCommon.filename
  End If

With graShockData
  .Visible = True
  With .Plot
    .Axis(VtChAxisIdX).ValueScale.Minimum = 0
    .Axis(VtChAxisIdX).ValueScale.Maximum = (graShockData.DataGrid.RowCount)
            * 0.05 / 1000
    .Axis(VtChAxisIdX).ValueScale.MajorDivision = 0.001
    .Axis(VtChAxisIdX).AxisTitle = "Time (ms)"
```

```
        .Axis(VtChAxisIdY).AxisTitle = "Acceleration (G's)"
        .UniformAxis = False
        .LocationRect.Min.Set 0, 0
        .LocationRect.Max.Set graShockData.Width - 450, graShockData.Height - 450
    End With
End With

frmMain.MousePointer = vbDefault
End Sub
'=============================================================

Private Sub txtFn_GotFocus()
    txtFn.SelStart = 0
    txtFn.SelLength = Len(txtFn.Text)
End Sub
'=============================================================

Private Sub txtFn_KeyPress(KeyAscii As Integer)
    If KeyAscii = 8 Then 'allow backspacing
        Exit Sub
    End If

    'only allow digits to be entered
    If KeyAscii < 48 Or KeyAscii > 57 Then
        If KeyAscii = 13 Or KeyAscii = 9 Then txtSampleFn.SetFocus  '13 is Return, 9 is
```

```
                Tab
    KeyAscii = 0 'ignore all other keystrokes
    Beep
  End If

  'criteria for moving focus to next text box
  If Len(txtFn.Text) >= txtFn.MaxLength And txtFn.SelLength <> Len(txtFn.Text)
      Then
    'txtDampRatio.SetFocus
    txtSampleFn.SetFocus
  End If
End Sub
'=============================================================

Private Sub txtSampleFn_GotFocus()
  txtSampleFn.SelStart = 0
  txtSampleFn.SelLength = Len(txtSampleFn.Text)
End Sub
'=============================================================

Private Sub txtSampleFn_KeyPress(KeyAscii As Integer)
  If KeyAscii = 8 Then 'allow backspacing
    Exit Sub
  End If
'only allow digits to be entered
```

```
   If KeyAscii < 48 Or KeyAscii > 57 Then
      If KeyAscii = 13 Or KeyAscii = 9 Then txtX0.SetFocus  '13 is Return, 9 is Tab
      KeyAscii = 0 'ignore all other keystrokes
      Beep
   End If

   If Len(txtSampleFn.Text) >= txtSampleFn.MaxLength And txtSampleFn.SelLength
         <> Len(txtSampleFn.Text) Then
      txtX0.SetFocus
   End If
End Sub
'=============================================================

Private Sub txtX0_GotFocus()
   txtX0.SelStart = 0
   txtX0.SelLength = Len(txtX0.Text)
End Sub
'=============================================================

Private Sub txtX0_KeyPress(KeyAscii As Integer)
If KeyAscii = 8 Then 'allow backspacing
      Exit Sub
   End If
'only allow digits to be entered
   If KeyAscii < 48 Or KeyAscii > 57 Then
```

```
    If KeyAscii = 13 Or KeyAscii = 9 Then txtX1.SetFocus  '13 is Return, 9 is Tab
    If KeyAscii = 46 Then Exit Sub  'decimal
    KeyAscii = 0 'ignore all other keystrokes
    Beep
  End If
If Len(txtX0.Text) >= txtX0.MaxLength And txtX0.SelLength <> Len(txtX0.Text)
      Then
    txtX1.SetFocus
End If
End Sub
'=============================================================
Private Sub txtX1_GotFocus()
  txtX1.SelStart = 0
  txtX1.SelLength = Len(txtX1.Text)
End Sub
'=============================================================
Private Sub txtX1_KeyPress(KeyAscii As Integer)
If KeyAscii = 8 Then 'allow backspacing
    Exit Sub
  End If
'only allow digits to be entered
  If KeyAscii < 48 Or KeyAscii > 57 Then
  If KeyAscii = 13 Or KeyAscii = 9 Then txtFn.SetFocus  '13 is Return, 9 is Tab
```

```
    If KeyAscii = 46 Then Exit Sub  'decimal
        KeyAscii = 0 'ignore all other keystrokes
        Beep
    End If
If Len(txtX1.Text) >= txtX1.MaxLength And txtX1.SelLength <> Len(txtX1.Text) Then
        txtFn.SetFocus
    End If
End Sub
```

Splash Screen:

```
Option Explicit
Private Sub Form_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then
        frmMain.Show
        Unload Me
    End If
End Sub

Private Sub Form_Load()
    lblVersion.Caption = "Version " & App.Major & "." & App.Minor & "." &
            App.Revision
End Sub
```

```
Private Sub Frame1_Click()
   frmMain.Show
   Unload Me
End Sub

Private Sub tmrTimer_Timer()
   frmMain.Show
   Unload Me
End Sub
```

Code Module:

```
Option Explicit
Type ShockData
   sngMS(2000) As Single
   sngG(2000) As Single
End Type
```

REFERENCES

REFERENCES

1. Newton, Robert E. "Fragility Assessment Theory and Test Procedure." U.S. Naval Postgraduate School.

2. Selected ASTM Standards on Packaging. Fourth Edition. Philadelphia, PA: ASTM, 1994.

3. Henderson, George. "Advanced Techniques For Shock and Vibration Analysis As Applied To Distribution Engineering." Fall 1991 IOPP P2C2 Boston Meeting, IBM Corporation Shock and Vibration Seminars, 1987-91, Distribution Dynamics Labs Advanced Shock and Vibration Seminars, 1991.

4. Schwartz, H.B. and J. Santos. "Shock Response Spectrum (SRS): Analysis, Interpretation and Application." Poughkeepsie, NY: IBM Technical Report, 1989.

5. Henderson, George. "Improved Protective Package Drop Test Data Analysis." IBM Annual Shock and Vibration Seminar. Boca Raton, FL. IBM, 19 October 1990.

6. Daum, Matthew P. "Application of the Shock Response Spectrum to Product Fragility Testing." Masters Thesis. Michigan State University, E. Lansing, MI, 1994.

7. Burgess, Gary J. "Product Fragility and Damage Boundary Theory." Packaging Technology and Science Vol. I 5-10, 1988.

8. Burgess, Gary J. "Effects of Fatigue on Fragility Testing and the Damage Boundary Curve". Journal of Testing and Evaluation Vol. 24, No. 6, 1996.

9. Mendelson, Alexander. Plasticity, Theory and Application. New York, NY: Macmillan, 1968.

10. Callister, William D. Jr. Materials Science and Engineering An Introduction. Fourth Edition. New York, NY: John Wiley and Sons, 1997.

11. Collins, Jack A. Failure of Materials in Mechanical Design. Second Edition. New York, NY: John Wiley and Sons, 1993.

12. Burgess, Gary J. Lecture notes, Advanced Packaging Dynamics. Michigan State University, E. Lansing, MI.

13. Wiebull, W. Fatigue Testing and the Analysis of Results. New York, NY: Pergmon Press, 1961.

14. Singh, S. "Effect of Multiple Drops on the Damage Boundary Curve." Masters Thesis, Michigan State University, E. Lansing, MI, 1983.

15. Flugge, Wilhelm. Viscoelasticity. Waltham, MA: Blaisdell Publishing Company, 1987.

16. Bhattacharyya, G. and Richard Johnson. Statistical Concepts and Methods. New York, NY: John Wiley and Sons, 1977.

17. Ostrem, Fred E. and Godshall, W.D. "An Assessment of the Common Carrier Shipping Environment." Techincal Report FPL 22. Forest Products Laboratory, Forest Service, USDA, Madison, WI, 1979.

18. Giacin, Jack. Lecture notes, Advanced Polymer Materials. Michigan State University, E. Lansing, MI.

19. Hibbeler, R.C. Engineering Mechanics Dynamics. New Jersey: Prentice-Hall, Inc., 1995.

20. Marcondes, J., Waldeck, J., Burgess, G., and Singh, S. "Application of High-speed Motion Analysis to Measure Shock in Cushioned Drops." Packaging Technology and Science Vol. 3 51-55, 1990.