AN ABSTRACT OF THE THESIS OF

Seung-uk Ko for the degree of Master of Science in Mechanical Engineering presented on April 16, 2001. Title: <u>Computer-Aided Model Generation for High Performance Dynamometers</u>.

Redacted for Privacy^{d for Privacy}

Abstract approved: _____

S̆wǎvik A. Spiewak

A model has been developed for accurate emulation of the complex behavior of high performance dynamometers used in a broad range of applications including manufacturing systems, biomechanics research, and crash testing. The key objective of this research has been to reach the proper balance between the accuracy of emulation and complexity of the model and user-friendliness of the modeling procedure. As a result, a versatile, step-wise computer aided methodology was developed. The research focused further on the development and validation of efficient procedures and analysis techniques that allow rapid identification of distinctive dynamic relationships in force sensors and encapsulating these relationships in analytical, constitutive models.

The assumption of rigid body, lumped parameter nature of the modeled class of dynamometers underlie the methodology used in this research. This assumption is justified since the dominant source of the system's dynamic behavior is the elastic coupling between individual components, rather than deformations of the components themselves. Lagrange's energy formalism and *Mathematica's* symbolic programming environment, which facilitates arbitrary precision computations, are utilized for model generation.

Experimental analysis was implemented to verify the developed model. The actual motion of the dynamometer components caused by external excitation forces was studied and compared with responses predicted by the model. Spatial dis-

placements were obtained by analyzing acceleration signals from several sensors placed in suitable locations on the tested system. To enhance and speed up the validation of modeling results, a visualization module was developed. It allows rapid comparison and rigorous analysis of the spatial motion recorded from the actual system with the model based prediction.

COMPUTER AIDED MODEL GENERATION FOR HIGH
PERFORMANCE DYNAMOMETERS

By

Seung-uk Ko

A THESIS

Submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented April 16, 2001
Commencement June 2001

Master of Science thesis of Seung-uk Ko presented on April 16, 2001

APPROVED:

Redacted for Privacy

Major Professor, representing Mechanical Engineering

Redacted for Privacy

Chair of Department of Mechanical Engineering

Redacted for Privacy

Dean of Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Redacted for Privacy

Seung-uk Ko, Author

ACKNOWLEDGEMENT

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF APPENDICES

(XYZ) = global coordinate system comprises of $X, Y,$ and Z axes

$(XYZ)_A$ = coordinate system at the plate's center mass comprises of $X_A, Y_A,$ and Z_A axes

$(XYZ)_I$ = instantaneous coordinate system of the plate comprises of $X_I, Y_I,$ and Z_I axes

$(XYZ)_R$ = reference coordinate system of the plate comprises of $X_R, Y_R,$ and Z_R axes

a_C, a_P = accelerations of point C and P, respectively, in $(XYZ)_R$

a_{cal} = calculated acceleration

$a_{i,j}$ = acceleration in i direction at at corner j; i=x, y, and z; j=C, 1, 2, and 3

a_s = acceleration signal from sensor

$\mathbf{B}_{SDE}$ = matrix of damping of SDE

$b_{pi,j}$ = damping constants of SDE connected between plate and base in location j with i direction; i=x, y, and z; j=1, 2, 3, and 4

C = origin of $(XYZ)_I$

C_R = origin of $(XYZ)_R$

c_s = calibrated factor [V/g] (sensitivity)

$\mathbf{D}_i^j$ = 4×1 matrix describes a vector from point i to point j

$\mathbf{d}_C, \mathbf{d}_P$ = generalized coordinates of point C, and P, respectively, with respect to (XYZ)

$\mathbf{d}_I$ = generalized coordinate of point C with respect to $(XYZ)_R$

$\mathbf{d}_O$ = generalized coordinate of point C_R with respect to (XYZ)

F_i = input force in i direction

G = system's center of mass

G_{ij} = transfer function of i direction from the j direction input

$G_D(z)$ = discrete-time transfer function

G = acceleration due to gravity

$\mathbf{I}_{i,r}$ = inertia matrix

i = index denoting the direction

$\mathbf{K}$ = vector of magnification coefficients

$k_{pi,j}$ = spring constants of SDE connected between plate and base in location j with i direction; i=x, y, and z; j=1, 2, 3, and 4

L = Lagrangian function

m_i = moment about i axes

m_P, m_B = mass of plate, mass of base

$\mathbf{O}$ = origin of the (XYZ)

$\overline{\mathbf{P}}$ = position vector for the location of SDE

P = arbitrary point on a rigid plate

Q_i = vector of external force and torque associated with i-th generalized coordinate

$\mathbf{q}_i$ = vector of generalized coordinate for i-th

$\dot{\mathbf{q}}_i$ = time derivative vector of generalized coordinate for i-th

r_x, r_y, r_z = distances between accelerometers

s_v = the sensitivity of accelerometer

T = sampling period

T = total kinetic energy

T_i = total kinetic energy of i-th body

U = total potential energy

U_i = total potential energy of i-th body

$\mathbf{T_{HT}}$ = total homogeneous transformation matrix

$\mathbf{T_P}$ = 'Pitch' transformation matrix

$\mathbf{T_R}$ = 'Roll' transformation matrix

$\mathbf{T_T}$ = 'Translational' transformation matrix

$\mathbf{T_Y}$ = 'Yaw' transformation matrix

TMP = transformation matrix of the platform

TMB = transformation matrix of the base

TF = transfer function

U(s) = input signal in s domain

V_o = amplifier output voltage

x, y, z = translations of point G relatively to point O, parallel to X, Y, and Z axes

x_C, y_C, z_C = translations of point C relatively to point O, parallel to X, Y, and Z axes

x_G, y_G, z_G = translations of point G relatively to point O, parallel to X, Y, and Z axes

x_I, y_I, z_I = translations of point C relatively to point C_R, parallel to X, Y, and Z axes

$Y(s)$ = output signal in s domain

α = angular accleration of point P in $(XYZ)_I$

α_i = angular accelerometer component of vector α ; i = x, y, and z

Δt = sampling period

θ, ϕ, ψ = rotations of $(XYZ)_G$ around X, Y, and Z axes, respectively

θ_C, ϕ_C, ψ_C = rotations of $(XYZ)_I$ around X, Y, and Z axes, respectively

θ_I, ϕ_I, ψ_I = rotations of $(XYZ)_I$ around X_R, Y_R, and Z_R axes, respectively

θ_O, ϕ_O, ψ_O = rotations of $(XYZ)_R$ around X, Y, and Z axes, respectively

ϖ = angular velocity of point P in $(XYZ)_I$

ϖ_i = angular velocity component of vector ϖ ; i = x, y, and z

ζ = damping coefficient

COMPUTER-AIDED MODEL GENERATION FOR HIGH PERFORMANCE DYNAMOMETERS

1. INTRODUCTION

Model derivation has many important aspects for understanding various dynamic systems. Properly emulated dynamic models not only provide the information of a system without troublesome experimental analysis, but also make it possible to predict the response of the system given certain excitation input. The dynamic properties of a system should be carefully considered during the modeling procedure to make it closer to the real system. Also this modeling should not be too complicated so that it is manageable for any analysis. Model derivation needs modification to make is closer to the real system or to make it a simpler one. For these kinds of modification, there should be a credible standard to for comparison. Experimental analysis is necessary for that comparison. Proper model derivation of a system base on comparison with experimental analysis is a main goal of this research.

1.1 MODEL GENERATION AND VALIDATION

Although more computation is needed for symbolic manipulation, the benefits are worth it. First, once the symbolic program is developed, it can be easily modified without re-modeling for different configurations. Second, it provides almost infinite precision of calculation because it uses symbols instead of numbers. And third, it gives an intuitive idea about the program because of symbols.
In the modeling of a dynamometer, a rigid body assumption is applied. The dynamometer consists of two rigid plates connected with each other with spring and damping elements. Lagrange's energy formalism is implemented to obtain the

equations of motion. For ease of computation, linearization procedure is applied with the small angular rotation assumption.

Experimental analysis should be carefully considered because is provides good validation of the model. The signal procedure, that is necessary for the calculation of displacements from the raw signals, is taken with deliberate steps.

1.2 SCOPE OF WORK

The system, which for this research is dynamometer, needs to be simplified to be emulated in a manageable way. Thus the model in this research is considered as rigid body motion. Flexible body motion is beyond the scope of this research, so vibration of flexible mode should not be a dominant factor to expect reasonable result from the model of system.

Laplace transform is applied to get transfer function from the differential equations of motions. The reason to use Laplace transform instead of state variable is to reduce the amount of computation.

1.3 CHAPTER OVERVIEW

In Chapter 2, distributed parameter system and lumped parameter system are presented as common methodologies. And three different formalism techniques are introduced. The advantages and disadvantages of using symbolic and numeric models are also discussed in this chapter.

The methodology and formalism used in this research are generally discussed in Chapter 3. In Chapter 4, the actual model derivation of dynamometer is taken into account. Several techniques for experimental validation are presented in Chapter 5. Signal processing procedures are carefully considered for the signals captured from the accelerometers, and the results of these procedures are compared with the results of the simulation of the model.

Visualization of excited dynamometer is included in Chapter 6 as an application of this research. In Chapter 7, conclusion of this research and the way to go in the future research are outlined.

2. LITERATURE REVIEW

In each procedure for CAMGHD[1], there can be many different ways to approach. Each method has its own advantages and disadvantages. Decisions such as whether to use symbolic or numeric computation, distributed and lumped systems, or linear or nonlinear calculation should be chosen so that the developed model satisfies the required conditions.

This chapter outlines the structure configuration and the theories of dynamics used in this research.

2.1 BUILDING MODELS

It is important to note that no system can be modeled exactly; inclusion of all the parameters affecting a particular system would be impossible to construct and analyze [Karnopp and Rosenberg, 1975]. On the contrary, if the model is too simple by too much simplification, it will not represent the real system. The desired model of system should be one that is manageable, but also one that includes the most important information about the system.

The first step for modeling is to disassemble the real system into components in terms of dynamics. Distributed parameter system or lumped parameter system can be considered for this first step.

If flexible components make up the system, using a distributed parameter approach can be a good decision. And a lumped parameter approach is a good decision for systems with rigid components. Also, the combination of these two approaches can be applied.

---

[1] CAMGHD : Computer-Aided Model Generation of a High performance Dynamometer
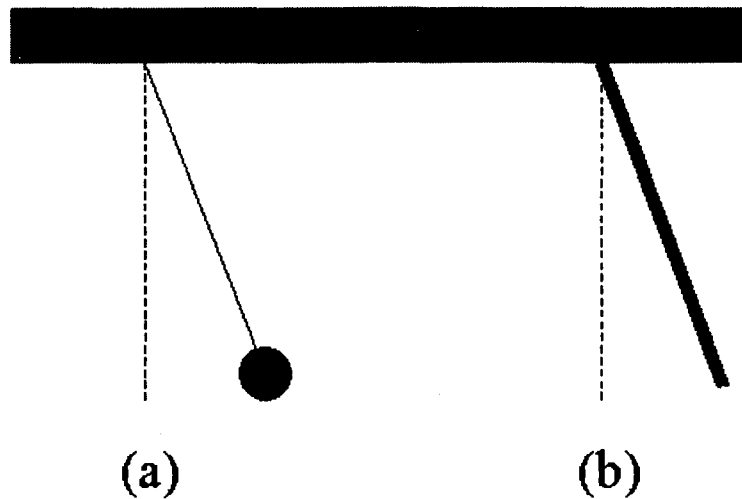
Figure 2.1: Pendulums with (a) lumped and (b) distributed masses.

2.1.1 Distributed Parameter System

Distributed or continuous parameter method can be applied for the dynamic system where the flexible modes of a particular body are significant. The distributed parameter system typically is a better representation of real system. However it also typically requires greater computation compared with lumped parameter system..

Since these structures are truly "continuous", they possess an infinite number of degrees of freedom [Tomson, 1981]. For example, exciting a simply supported flexible machine tool spindle with a continuous mass and elasticity distribution can result in any of an infinite number of mode shapes.

Although the use of partial differential equations provides an excellent description of the system, they do not always produce obtainable results from controls theory for complex shapes or multiple bodies in the system. However, since in most cases the dominant modes are the lowest few, these modes shapes can be approximated by a polynomial fit for the spindle deflection [Ewins, 1984]. The result is a set of ordinary differential equations in place of a set of partial differential equations [Shabana, 1991].

The next question is which polynomial to use. Indeed, as structural shapes increase in complexity, the choice of polynomial fit becomes obscure. This problem can be remedied by the use of Finite Element methods. With these methods the structure is divided into simpler elements, and the deformations within each element are described by interpolating polynomials [Shabana, 1991]. These methods are often successfully implemented with good accuracy where body deformations are of a concern, but those require more intensive computation [Carne, et. Al., 1988; Cheung and Leung, 1991; Fagann, 1992; Friswell and Mottershead, 1995; Weaver and Johnston, 1987; Gysin; Zatarain, 1998; for spindles, Reddy and Sharan, 1987; Comparin, 1983; for machine tools, Bianchi and Paolucci, Weck, 1984, Brisbone, 1998].

Figure 2.2: Simply supported fixed & free flexible spindle with excited first mode.

2.1.2 Lumped Parameter System

In many cases, systems do not have to be considered as distributed parameter method if the deformation of bodies within the system is not a significant factor compared with dynamic behaviors of the structure. For these systems, the elastic couplings between individual components are the dominant dynamic factors. And each component of the system is considered a rigid body.

One example of such a structure is the spindle housing system shown in Fig. 2.3 [Aini, et. Al., 1990;Matsubara, 1988;Shin, et. al., 1990; Spiewak, 1995; Weikert, et. al., 1997]. In machining processes, low to medium frequency dynamics

of the structure play a critical role in tool path errors. This can be successfully modeled by the use of lumped parameters, since the housing and spindle structure are of sufficient rigidity such that their flexible modes (usually high frequency) have little influence on the dynamic frequency range of interest [Comparin, 1983; Weck, 1984; Weikert, et. al., 1998; Brisbone, 1998].
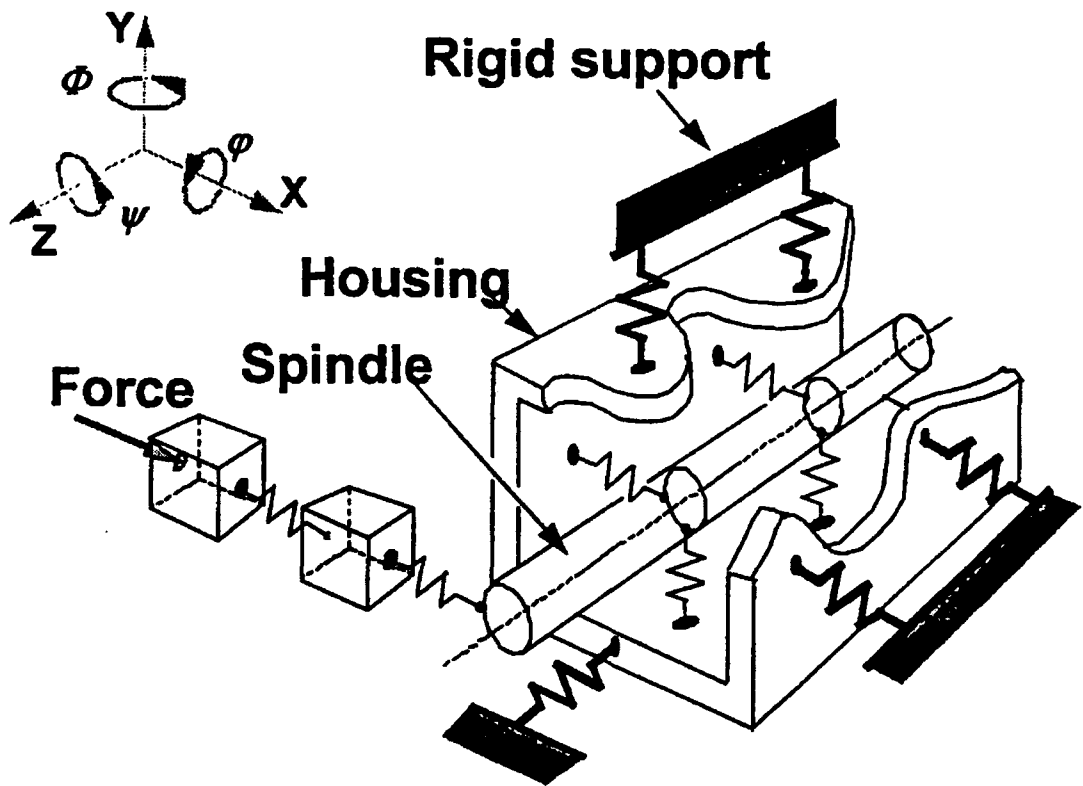
Figure 2.3 Rigid body model of spindle housing structure [Spiewak, 1995].

A pair of bearings, which are considered as springs and dampers, couples the spindle to the housing. Because the mass of the bearings is small compared to the spindle and housing, omission of these masses will not affect the results of the model. By that omission, the computation for model generation can be greatly reduced.

Advantages of the lumped parameter method include a reduced number of generalized coordinates, use of ordinary differential equations, simplified computations, and the existence of an obtainable result.
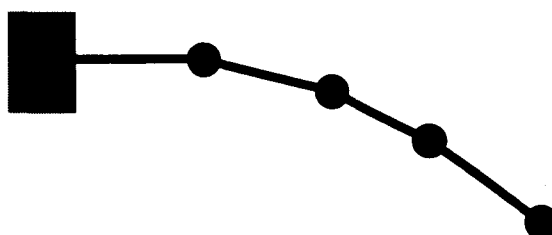
Figure 2.4 Rigid body approximation of fixed & free spindle mode shape.

In the case where there exists both dominant flexible mode and rigid body motion, it is useful to combine both methods [Weck, 1984]. In addition, formulation of the equations of motion for deformable bodies often finds it convenient to separate out the rigid body and deformational contributions from the overall motion [Ginsberg, 1995; Marion and Thornton, 1988; Weck, 1984].

2.2   MODELING METHODOLOGIES

There are three widely used formalisms: Newton [Marion and Thornton, 1988], Lagrange [Ginsberg, 1995; Marion and Thornton, 1980; Scheck, 1994], and Kane [Kane and Levinson, 1985; Kane, et. al., 1983]. Each of these three methodologies have their own advantages to use in different structure configuration.

Newton's Laws of Mechanics is the most accepted method for modeling systems. Newton's Second Law is used to obtain the ordinary differential equations of motion from a certain system. This is the most straightforward and intuitive way

for modeling and for the verification of the already developed models. For the system with simple structure, this is still an adequate method, but for the complex system, this is not a viable method to use.

Lagrange's method is a suitable for structures with increased complexity. Contrary to Newton's method, which is concerned with forces and torques, the Lagrange's method considers the energies (kinetic, potential and dissipative) of the system. Although more abstract, the generated equations are nearly identical to Newton's approach only in a slightly different form [Rosenthal and Sherman, 1986]. Defining the energies of a particular system is much easier than defining the forces and torques of a system. Thus this research uses Lagrange's energy method.

Kane's method deals with generalized active and inertia forces [Kane, et. al., 1983]. Using the cancellation of forces that contribute nothing on the body of a system, simplified equations can be derived. This is the most compact form with the easiest way to obtain the equations of motion. But there is also a set of associated kinematical equations that must be satisfied when using this method [Ginsberg, 1995].

Lagrange's energy formalism was chosen in this research for the sake of convenience of using symbolic problem-solving environment provided by *Mathematica*.

2.3 NUMERIC VS. SIMBOLIC MODELS

As used in early-automated modeling, a numeric method can be free from intensive computations. But it also has several drawbacks, including:

1. Repeated setup of the dynamic equations at each computation step or integration, resulting in excessive operations and extensive computation time.
2. Difficulty in implementing control strategies in numerical equations, obstructing real time operations as required by some multi-body systems.

3. Unclear physical insight into the system as a result of numerical expressions.

4. Equations of motion existing as only mathematical operations in the computer program [Lieh and Haque, 1991; Hale and Meirovitch, 1978].

Numeric algorithms can give accurate results, but not sufficient for the reasons mentioned above. The fascinating advantages of using symbolic method include:

1. Infinite precision, since calculated values are not subjected to accumulated errors caused by limited machine precision.

2. One time model derivation, since iterative calculations only involve parameter value substitutions.

3. Clear intuitive insight into the physical system.

4. Straightforward control strategy implementation as a result of (3).

5. Greater accuracy of estimating unknown parameters.

6. Ability to potentially produce closed form solutions, as opposed numeric computations that give iterative solutions [Brisbone, 1998].

2.4   MODEL LINEARIZATION

Few physical elements of a system in nature display truly linear characteristics. Typically, the equations of motion for dynamic systems are nonlinear. Such equations are much more difficult to solve than linear ones, and the kinds of possible motions resulting from the nonlinear model are much more difficult to categorize than those resulting from the linear model [Gene F. Franklin, 1994]. The more important difference is the behaviors of the linear and nonlinear models. The behavior of a linear model can be understood much more comprehensively and with a small fraction of the effort required to analyze a

nonlinear model. As a result, an important aspect of any modeling program is effective and accurate linearization of the system where appropriate.

Several attempts have been done to make linearization more efficient. Miller and White used an innovative approach by writing all transformation matrices as exponentials, making differentiation and thus linearization easier [Miller and White, 1987].

For most systems, the movement of interest usually involves small displacements or rotations about a nominal or equilibrium position. This nominal position is not necessarily fixed, but can change with varying configurations of the system. For such a system, the most widely used method of linearization is a multi-variable Taylor Series expansion about the nominal position [Ginsberg, 1995; Marion and Thornton, 1988]. Some that is done to perform the expansion on the complete nonlinear equations of motions, while others perform the expansion at an earlier stage of equation development. For the Lagrange's energy formalism, the simplest form of linearization is accomplished by expanding the energies, which is also done in this work. Another advantage that arises from the expansion of potential energy is a pre-check concerning verification of the model, which is explained in the next chapter.

2.5   CLOSURE

Structure configuration and dynamical theories are introduced. The comparison of distributed and lumped parameter system provides a guide for choosing proper, modeling methods of a system. Widely used formalisms such as Newton's, Lagrange's, and Kane's method, are mentioned. The advantage of numeric method provides the reason of using numeric method in this research. Linearlization, which is essential for manageable computation is also mentioned. In next chapter, procedures of derivation of the model are described.

3. COMPUTER AIDED MODEL GENERATION

For the complete understanding of a certain dynamic system, having various analysis approaches with different method, would be useful, or might be necessary. For example, methods such as simulated response to actual input, location of poles and zeros, test for the controllability, and modal properties of system could be considered. To perform these kinds of analysis, different forms of model such as the transfer-function form, or the state-variable form are required. Also it should be possible for the model to be converted in different domains including continuous time domain, Laplace domain, or discrete-time domain. General concept of modeling is described in this chapter.

3.1 GENERALIZED RIGID BODY SYSTEM

Lumped parameter method is applied in this research to represent the dynamic relationships between forces and displacements. The components of each body are connected by 'spring and damper element' (SDE). The bearing couples such as shown in fig. 2.3, can be an example of system represented by SDE. Each rigid body has six degrees of freedom (DOF). Three DOF from translations, and other three DOF from rotations. So If there are 'n' number of rigid body, whole DOF will be 6n.

Efficient mapping between the actual system's components and their representation in the model is possible by the method mentioned above
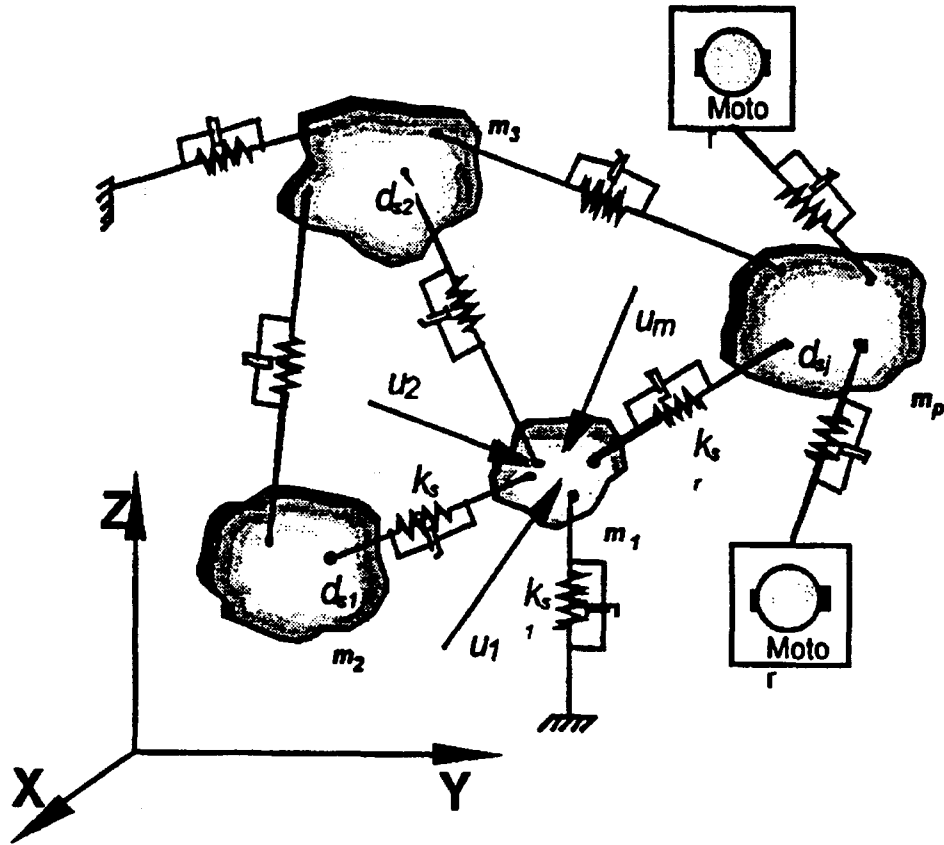
Figure 3.1 Generalized multi-degree-of-freedom rigid body system [Spiewak,1995].

3.2 LAGRANGE'S ENERGY FORMALISM

After defining the generalized model structure, the equations of motion can be derived. Since defining the energies of arbitrary structure is generally simpler than defining the forces, Lagrange's Energy Formalism is used in this research.

For the conservative system, the Lagrange's equations can be stated as

$$\frac{\partial}{\partial t}\frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_j} = 0 \qquad (3.1)$$

The choice for L is not unique, but the natural choice (and the convention followed here) is to set

$$L = T - U \tag{3.2}$$

where T represents the kinetic energy and U represents the potential energy.

External forces acting on the system are taken into account by Lagrange's equations of the second kind [Pandit, 1991], and shown as below

$$\frac{\partial}{\partial t}\frac{\partial L}{\partial \dot{q}_j} - \frac{\partial L}{\partial q_j} = Q_i \tag{3.3}$$

where Q_i represents the external forces or torques associated with the i-th generalized coordinate.

3.3   ASSOCIATED SYSTEM ENERGIES

Instead of considering force equilibriums for the direct derivation of equation of motion from a structure, in Lagrangian method, a concentration on energies of a structure is taken. The kinetic, potential, and damping (dissipation) energies are the only of concern. Fig.3.1 shows the dashpots and springs which are main sources of all energies of the structure.

3.3.1   Kinetic Energy

In assumption that SDEs are massless, the only contributors to the kinetic energy are the movements of the rigid bodies themselves. So the kinetic energy can be separated into two parts, namely translational, and rotational.

$$T_i = T_{i,trans} + T_{i,rot} \tag{3.4}$$

the translational kinetic energy is written as

$$T_{i,trans}(\dot{\mathbf{q}}_{i,T}) = \frac{1}{2}\dot{\mathbf{q}}_{i,T}\mathbf{m}_i\dot{\mathbf{q}}_{i,T} \qquad (3.5)$$

where $\mathbf{m}_i$ is mass matrix in diagonal form and $\dot{\mathbf{q}}_{i,T} = \{\dot{X}_i, \dot{Y}_i, \dot{Z}_i\}$ is the generalized translational vector of velocity for the i-th rigid body in the global reference frame.

The generalized rotation vector of velocity for the i-th rigid body is also defined as $\dot{\mathbf{q}}_{i,r} = \{\dot{\theta}_i, \dot{\phi}_i, \dot{\varphi}_i\}$, as it is for translational vector of velocity. The rotational kinetic energy is written as

$$T_{i,rot}(\dot{\mathbf{q}}_{i,r}) = \frac{1}{2}\dot{\mathbf{q}}_{i,r}^T \mathbf{I}_{i,r}\dot{\mathbf{q}}_{i,r} \qquad (3.6)$$

The inertia tensor, $I_{i,r}$, will always be diagonal as long as the local coordinate axes correspond with the principal axes of inertia [Marion and Thornton, 1988; Ginsberg, 1995] for the body. This is not always the case, but it is generally less tedious to define the principal inertias (diagonal elements of the tensor) and transform to another configuration rather than fill in all the elements of the tensor for each change in orientation.

By defining transformation matrix λ_i, the rotations about global axes can be easily obtained from the rotation about the local axes. Especially, more than two rigid bodies are concerned, the rotation about global axes is useful.

$$\mathbf{q}_{i,R} = \lambda_{i,R}\mathbf{q}_{i,r} \qquad (3.7)$$

And for the sake of clarity, the transformation matrix λ_i is assumed as time independent, so the rotational velocity about the global axes is obtained in same way.

$$\dot{\mathbf{q}}_{i,R} = \lambda_{i,R}\dot{\mathbf{q}}_{i,r} \qquad (3.8)$$
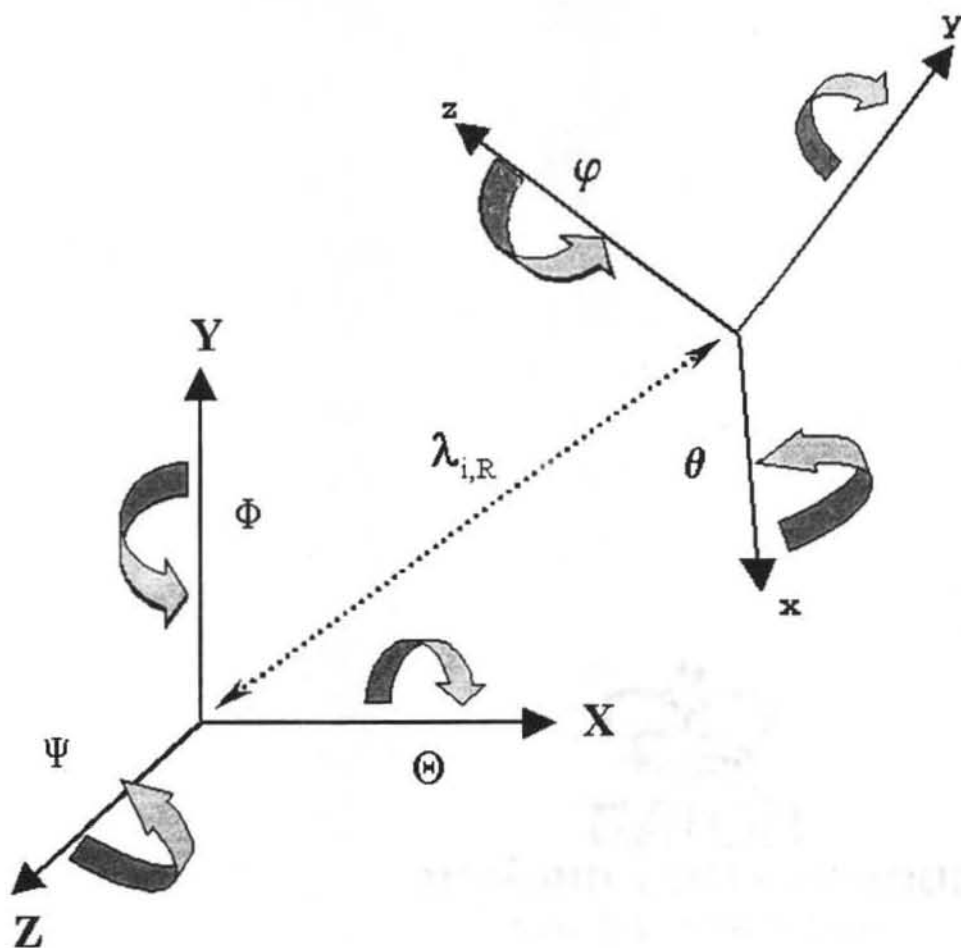
Figure 3.2 Transformation, $\lambda_{i,R}$, relating rotations between local and global coordinates.

Substituting Eq.3.8 into Eq.3.6, the rotational kinetic energy for the i-th rigid in a global coordinates are obtained as

$$T_{i,rot}(\dot{\mathbf{q}}_{i,R}) = \frac{1}{2}\dot{\mathbf{q}}_{i,R}^{T}(\lambda_{i,R}^{-1})^{T}\mathbf{I}_{i,r}\lambda_{i,R}^{-1}\dot{\mathbf{q}}_{i,R} \qquad (3.9)$$

For the orthogonal matrices, the inverse matrices are equal to the transpose matrices. And the rotation matrix is one of the orthogonal matrices. So the Eq.3.17 can be simplified as

$$T_{i,rot}(\dot{\mathbf{q}}_{i,R}) = \frac{1}{2}\dot{\mathbf{q}}_{i,R}^T\lambda_{i,R}\mathbf{I}_{i,r}\lambda_{i,R}^T\dot{\mathbf{q}} \tag{3.10}$$

Substitution of Eq.3.5 and Eq.3.10 into Eq.3.4 gives the total kinetic energy for the i-th rigid body about global axes, and shown as

$$T_{i,rot}(\dot{\mathbf{q}}_{i,T},\dot{\mathbf{q}}_{i,R}) = \frac{1}{2}\dot{\mathbf{q}}_{i,T}^T\mathbf{m}_{i,T}\dot{\mathbf{q}}_{i,T} + \frac{1}{2}\dot{\mathbf{q}}_{i,R}^T\lambda_{i,R}\mathbf{I}_{i,r}\lambda_{i,R}^T\dot{\mathbf{q}}_{i,R} \tag{3.11}$$

The total kinetic energy for a system of n rigid bodies is simply the summation of the kinetic energies of each individual body

$$T = \sum_{i=1}^{n}T_i(\dot{\mathbf{q}}_{i,T},\dot{\mathbf{q}}_{i,R}) \tag{3.12}$$

3.3.2   Potential and Damping Energies

The model shown from the Fig.3.1 has two kinds of potential energies, namely due to gravitation and elongation of SDEs. Gravitational potential energy can be obtained by using the vertical displacement of the rigid body from the original position. As follows

$$U_{i,g} = m_i g Z_i \tag{3.13}$$

The second form of energy storage is in compression or tension of the SDEs between the bodies. The elongation of any elastic element between bodies i and j that is due to a motion of the i-th body can be written as

$$l_{i,k} = l_{i,k}(\mathbf{q}_i) \qquad k = 1,2,\ldots,m \qquad (3.14)$$

where m is the number of springs which connecting the two bodies, and $\mathbf{q}_i = \{X_i, Y_i, Z_i, \theta_i, \phi_i, \varphi_i\}$ is the coordinate vector including translations and rotations of the body i.

All elongations between the bodies due to the movement of the i-th body can be written as

$$\mathbf{L}_i(\mathbf{q}_i) = [l_{i,1}, l_{i,2}, \ldots l_{i,m}]^T \qquad (3.15)$$

If the coordinate of the j-th body is concerned in a same way, the potential energy between the i-th and j-th body due to the elongation of SDE between the two bodies can be calculated as

$$U_{ij}(\mathbf{q}_i, \mathbf{q}_j) = \frac{1}{2}[\mathbf{L}_i - \mathbf{L}_j]^T \mathbf{K}_{SDE}[\mathbf{L}_i - \mathbf{L}_j] \qquad (3.16)$$

where $\mathbf{K}_{SDE}$ represents the stiffness matrix which consists of stiffness constants of the SDEs which connect the two bodies. And by summing all potential energy of 'n' number of rigid bodies, the total potential energy is given as

$$U = \sum_{i=1}^{n}\left(U_{i,g} + \frac{1}{2}\sum_{j=1}^{n}U_{ij}(\mathbf{q}_i, \mathbf{q}_j)\right) \qquad (3.17)$$

The dissipation energy can be calculated in the similar manner to the elongation energy, except the velocity of deflection is used instead of the displacement

$$D_{ij}(\dot{\mathbf{q}}_i, \dot{\mathbf{q}}_j) = \frac{1}{2}[\dot{\mathbf{L}}_i - \dot{\mathbf{L}}_j]^T \mathbf{B}_{SDE}[\dot{\mathbf{L}}_i - \dot{\mathbf{L}}_j] \qquad (3.18)$$

where $\mathbf{B}_{SDE}$ represents a diagonal matrix of damping of the SDEs between the *i-th* and *j-th* bodies. Again, summation over all n bodies gives the total damping energy

$$D = \sum_{i=1}^{n} \sum_{i=j}^{n} D_{ij} (\dot{\mathbf{q}}_i, \dot{\mathbf{q}}_j) \qquad (3.19)$$

3.4   THE EQUATIONS OF MOTION

By substituting Eq.3.10 into Eq.3.11, the simplified result is obtained [Pandit, 1991] as

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}_i^g}\right) - \frac{\partial T}{\partial \mathbf{q}_i^g} + \frac{\partial U}{\partial \mathbf{q}_i^g} = Q_i \quad i = 1,2,\ldots,6n \qquad (3.20)$$

where Q_i represents the external force associated with the *i*-th generalized coordinate from the global list of generalized coordinate $\mathbf{q}^g$ representing all n bodies. Modifying this to include the damping energy is accomplished by the addition of another term
[Pandit, 1991]

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\mathbf{q}}_i^g}\right) - \frac{\partial T}{\partial \mathbf{q}_i^g} + \frac{\partial U}{\partial \mathbf{q}_i^g} + \frac{\partial D}{\partial \dot{\mathbf{q}}_i^g} = Q_i \quad i = 1,2,\ldots,6n \qquad (3.21)$$

For the multi-degree-of-freedom system, by solving Eq.3.21, the equations of a motion for each generalized coordinate is obtained. In *Mathematica* code[2] , Eq.3.29 is represented successfully by the following formula.

---

[2] The whole *Mathematica* code is attached in Appendix B

```
LagrEqns[T_, U_, Damp_, Q_List, Coor_List] :=
    Flatten[MapThread[
        {∂_t(∂_{∂,#2}T) - ∂_{#2}(T-U) + ∂_{∂,#2}Damp - #1 == 0} &,
        {Q, Coor}]];
```

To derive the equations of motion, the above function is provided with the kinetic energy, potential energy, dissipation energy, external forces, and coordinates of the system.

3.5   CLOSURE

General methodology of modeling by the use of Lagrange's energy method is discussed in this chapter. Properly defined energies with the appropriate coordinates provide the equations of motion by using *Mathematica* with the Control System Professional package. For the modeling of a complicated system including many rigid bodies, Lagrange's energy method gives a clear solution, because the only concerns in this method are energies rather than force-equilibriums in Newton's method.

4. MODELING OF DYNAMOMETER

In the structure of the dynamometer, a prismatic platform made of high strength steel is supported by the spring and damping elements (SDEs), as shown in Fig 4.1. Compared with the mass of the platform and base, the masses of the SDEs are negligible. So it is possible to assume the SDE as massless without affecting the result of analysis. Assuming that the platform and the base are rigid bodies, each of these has six degrees of freedom, three from translations and three from rotations. Considered as dynamic system, the dynamometer exhibits twelve resonance frequencies associated with its vibration modes [Chung, 1993].
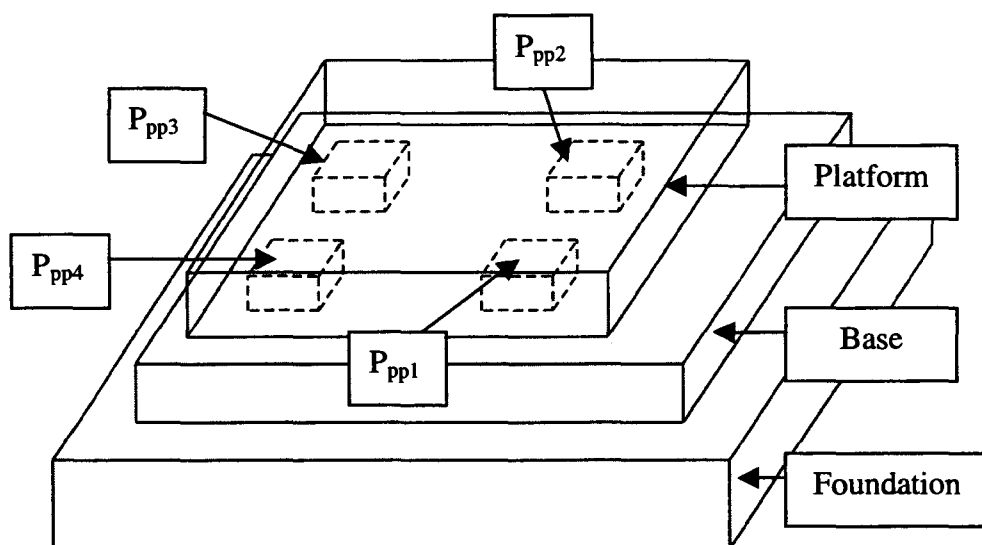
- P_{pp1}, P_{pp1}, P_{pp1}, P_{pp1} are the points on the bottom face of the platform, which are attached to SDEs.

Figure 4.1. Mechanical configuration of the dynamometer under consideration.[3]

---

[3] SDEs between the base and the foundation are omitted in the Fig.4.1.

4.1   TRANSFORMATION MATRICES

Twelve generalized coordinates are used for modeling of the dynamometer, namely six coordinates for the platform, and the other six coordinates for the base. Mass, inertia, stiffness and damping for each body are defined by suitable matrices as discussed in Chapter 3.

Combining rotational and translational motions of the platform and base, partial transformation matrices are derived. Next, by multiplying these transformation matrices, total homogeneous (4×4) transformation matrices are developed. For small angular displacements, the total transformation matrix of the platform is written as

$$TMP = \begin{pmatrix} 1 & -\psi 1^*[t] & \phi 1^*[t] & x1^*[t] \\ \psi 1^*[t] & 1 & -\theta 1^*[t] & y1^*[t] \\ -\phi 1^*[t] & \theta 1^*[t] & 1 & z1^*[t] \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad (4.1)$$

$x1^*[t]$, $y1^*[t]$, $z1^*[t]$:
  Translational coordinates of platform, incremental components only
$\theta 1^*[t]$, $\phi 1^*[t]$, $\psi 1^*[t]$:
  Rotational coordinates of platform, incremental components only

The total transformation matrix of base has the same structure while the coordinates are the translations and rotations are of the base.

$$TMB = \begin{pmatrix} 1 & -\psi 2^*[t] & \phi 2^*[t] & x2^*[t] \\ \psi 2^*[t] & 1 & -\theta 2^*[t] & y2^*[t] \\ -\phi 2^*[t] & \theta 2^*[t] & 1 & z2^*[t] \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad (4.2)$$

$x2^*[t]$, $y2^*[t]$, $z2^*[t]$:
  Translational coordinates of base, incremental components only
$\theta 2^*[t]$, $\phi 2^*[t]$, $\psi 2^*[t]$:
  Rotational coordinates of base, incremental components only

4.2 DEFINING POSITION VECTOR AND ENERGIES

The configuration of SDEs, as supporting elements between the platform and the base, and between the base and the foundation, define the position vectors of SDE's connection points on each with the platform and base as shown in Fig. 4.2 [Nickel, 1999].
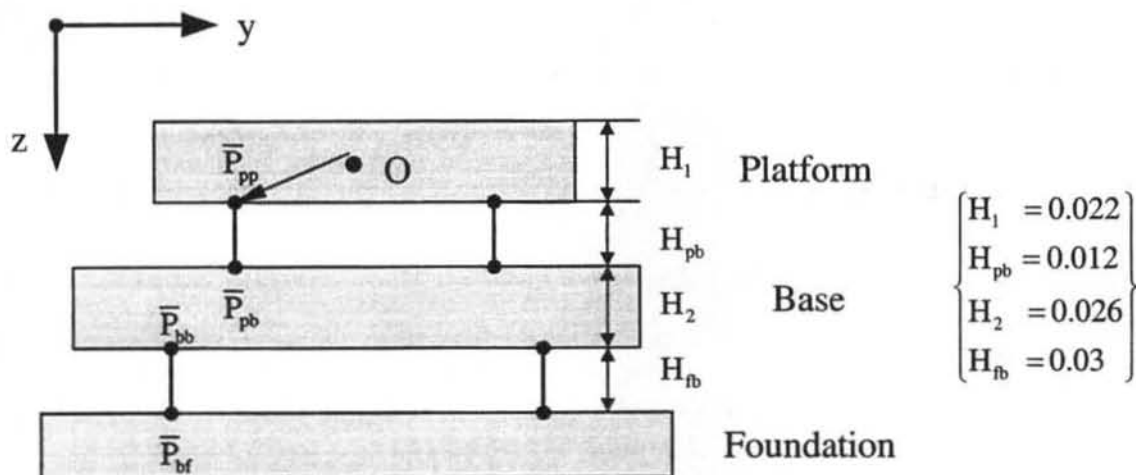
Figure 4.2 Simplified diagram of the Dynamometer configuration with Spring & Damping Elements (SDEs).

$\overline{P}_{pp1}$: Position vector of the platform-base connection[4], point #1 on the platform[5].

$\overline{P}_{pb1}$: Position vector of the platform-base connection, point #1 on the base.

$\overline{P}_{bb1}$: Position vector of the base-foundation, point #1 on the base.

$\overline{P}_{bf1}$: Position vector of the base-foundation, point #1 on the foundation.

Each individual position vector is defined as shown in Fig 4.2.

$\overline{P}_{pp1} = \{-a1, b1, h1, 1\}$

[4] the first subscript 'p' stands for the platform-base connection.

[5] the second subscript 'p', or 'b' stands for the point on the platform or on the base, respectively.

$$\overline{P}_{pp2} = \{a1, b1, h1, 1\}$$

$$\overline{P}_{pp3} = \{a1, -b1, h1, 1\}$$

$$\overline{P}_{pp4} = \{-a1, -b1, h1, 1\}$$
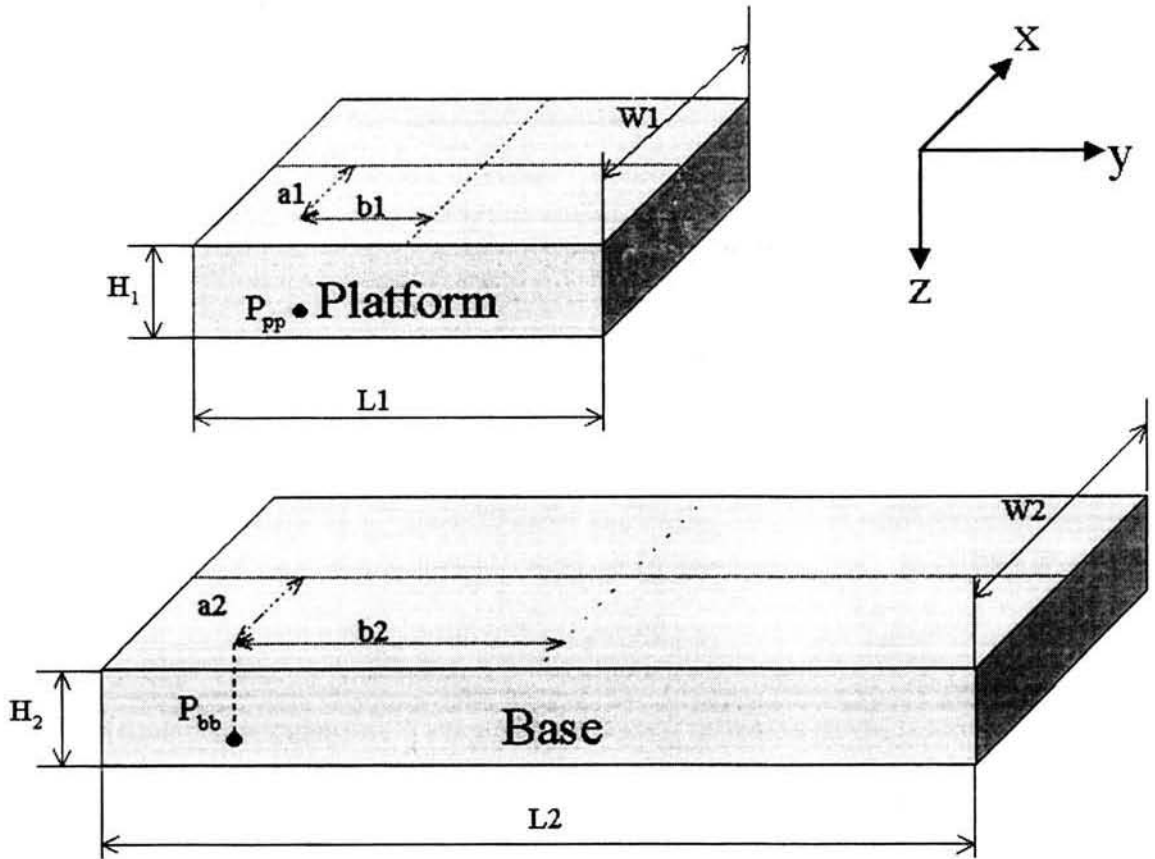
Figure 4.3. Configuration of the platform and the base [Dimensions are given in Appendix A.2].

Four position vectors on each body, such as Ppp1, Ppp2, Ppp3, and Ppp4 on the bottom face of platform, represent the position of each body in every moment and these points of position vectors are used to calculate all energies discussed in Chapter 3. For example, four position vectors on the bottom face of platform are expressed as a matrix in *Mathematica* as

VectorsOnPlate=Transpose[{Ppp1,Ppp2,Ppp3,Ppp4}];

MatrixForm[VectorsOnPlate]

$$\begin{pmatrix} -a1 & a1 & a1 & -a1 \\ b1 & b1 & -b1 & -b1 \\ h1 & h1 & h1 & h1 \\ 1 & 1 & 1 & 1 \end{pmatrix} \tag{4.3}$$

4.2.1   Potential Energy

Deflection caused by the rotational and translational motion of each body generates the major components of potential energy of the system, u, that is used for defining equations of motion by use of Lagrange's method. Deflections of the elements between the platform and the base, which are introduced in Eq.3.15 are calculated by the following expression.

$$(\mathbf{TMP} - \mathbf{I}) \mathbf{P}_{pp} - (\mathbf{TMB} - \mathbf{I}) \mathbf{P}_{pb} \tag{4.4a}$$

Which in *Mathematica* is expressed as

```
deflSpringBasePlatform=
  ((TMP- IdentityMatrix[4]).VectorsOnPlate -
    (TMB- IdentityMatrix[4]).VectorsOnTopOfBase)
```
$\tag{4.4b}$

The stiffness matrices of elements between the platform and the base, and between the base and the foundation, which are introduced in Eq.3.16 are defined in *Mathematica* as,

$$\text{stiffArrayOfBP} = \begin{pmatrix} k_{px1} & k_{px2} & k_{px3} & k_{px4} \\ k_{py1} & k_{py2} & k_{py3} & k_{py4} \\ k_{pz1} & k_{pz2} & k_{pz3} & k_{pz4} \\ 0 & 0 & 0 & 0 \end{pmatrix} ;$$

$$\text{stiffArrayOfBF} = \begin{pmatrix} k_{bx1} & k_{bx2} & k_{bx3} & k_{bx4} \\ k_{by1} & k_{by2} & k_{by3} & k_{by4} \\ k_{bz1} & k_{bz2} & k_{bz3} & k_{bz4} \\ 0 & 0 & 0 & 0 \end{pmatrix} ; \tag{4.5}$$

With the matrices of deflection and the matrices of stiffness defined above, the total potential energy of SDEs between the platform and the base designated as "TotalPotOfBP" is calculated in *Mathematica* as

```
res1 = Transpose[deflSpringBasePlatform] * stiffArrayOfBP *
   deflSpringBasePlatform;
Clear[i, j, ULSOfBP]
For[ULSOfBP = 0; i = 1, i ≤ 3, i++, For[j = 1, j ≤ 4, j++,
   TotalPotOfBP = ULSOfBP + 1 / 2 res1[[i, j]]]];
```
(4.6)

Similarly, the total potential energy of the SDEs between the base and foundation, "TotalPotOBF" is found in Mathematica as

```
res2 = Transpose[deflSpringBaseFoundation] * stiffArrayOfBF *
   deflSpringBaseFoundation;
Clear[i, j, ULSOfBF]
For[ULSOfBF = 0; i = 1, i ≤ 3, i++, For[j = 1, j ≤ 4, j++,
   ToTalPotOfBF = ULSOfBF + 1 / 2 res2[[i, j]]]];
```
(4.7)

Finally, the total potential energy in SDEs is found as

```
TotalSprPot = TotalPotOfBP + ToTalPotOfBF;
```
(4.8)

Adding gravitational potential energy to total spring potential energy, total potential energy is obtained according to Eq.3.17.

```
Ut = UG + TotalSprPot;
```
(4.9)

Where,

UG : ravitational Potential Energy
TotalSprPot : Total Elongation Potential Energy

4.2.2 Kinetic Energy

Derivatives of the generalized coordinates are used to calculate the kinetic energy. There are two components of this energy, namely due to the translational and the rotational generalized coordinates. By adding these two, total kinetic energy is obtained according to Eq.3.12, as

$$
\begin{aligned}
T_t \;=\; & T_{trans} + T_{rot} \\
& \frac{1}{2}\,\big(mP\,x1'[t]^2 + mP\,y1'[t]^2 + mP\,z1'[t]^2\big) \;+ \\
& \frac{1}{2}\,\big(mB\,x2'[t]^2 + mB\,y2'[t]^2 + mB\,z2'[t]^2\big) \;+ \\
& \frac{1}{2}\,\big(JP_{xx}\,\theta1'[t]^2 + JP_{yy}\,\phi1'[t]^2 + JP_{zz}\,\psi1'[t]^2\big) \;+ \\
& \frac{1}{2}\,\big(JB_{xx}\,\theta2'[t]^2 + JB_{yy}\,\phi2'[t]^2 + JB_{zz}\,\psi2'[t]^2\big)
\end{aligned}
\tag{4.10}
$$

4.2.3 Dissipation Energy

Damping constants of SDEs need to be combined into the matrices in order to calculate the total damping energy according to Eq.3.18. Damping elements between the platform and the base are defined in *Mathematica* as

$$
\text{dampArrayOfBP} =
\begin{pmatrix}
b_{px1} & b_{px2} & b_{px3} & b_{px4} \\
b_{py1} & b_{py2} & b_{py3} & b_{py4} \\
b_{pz1} & b_{pz2} & b_{pz3} & b_{pz4} \\
0 & 0 & 0 & 0
\end{pmatrix} ; \qquad
\text{dampArrayOfBF} =
\begin{pmatrix}
b_{bx1} & b_{bx2} & b_{bx3} & b_{bx4} \\
b_{by1} & b_{by2} & b_{by3} & b_{by4} \\
b_{bz1} & b_{bz2} & b_{bz3} & b_{bz4} \\
0 & 0 & 0 & 0
\end{pmatrix} ;
$$

DampArrayOfBP : damping coefficient matrix between the base and the platform
DampArrayOfBF : damping coefficient matrix between the base and the Foundation   (4.11)

With the matrices of deflection's derivatives and matrices of damping constants, total dissipation energy is calculated by *Mathematica* according to Eq. 3.18, as

```
res3 = Transpose[∂_t deflSpringBasePlatform] * dampArrayOfBP *
  ∂_t deflSpringBasePlatform;
```

```
Clear[i, j, DampOfBP]
```

```
For[DampOfBP = 0; i = 1, i ≤ 3, i++,
  For[j = 1, j ≤ 4, j++, DampOfBP = DampOfBP + 1/2 res3[[i, j]]]];
```

```
res4 = Transpose[∂_t deflSpringBaseFoundation] * dampArrayOfBF *
  ∂_t deflSpringBaseFoundation;
Clear[i, j, DampOfBF]
For[DampOfBF = 0; i = 1, i ≤ 3, i++,
  For[j = 1, j ≤ 4, j++, DampOfBF = DampOfBF + 1/2 res4[[i, j]]]];          (4.12)
```

```
TotalDampOfBP = DampOfBP;
TotalDampOfBF = DampOfBF;
D_t = TotalDampOfBP + TotalDampOfBF;
```

Where,

DampOfBP : Total damping energy between the base and the platform
DampOfBF : Total damping energy between the base and the Foundation

4.2.4   External Input Force and Moment

All necessary external input forces and moments in Eq.3.21 are contained in
the vector designated below as "minGen". Coordinates of the force application
points within the plate and the base needed to calculate moments generated by the
external forces about the center of gravity are included in "minGen".

```
MatrixForm[minGen]
```

$$
\begin{pmatrix}
f_x[t] \\
f_y[t] \\
f_z[t] \\
-z_{in} f_y[t] + y_{in} f_z[t] + m_x[t] \\
z_{in} f_x[t] - x_{in} f_z[t] + m_y[t] \\
-y_{in} f_x[t] + x_{in} f_y[t] + m_z[t] \\
f_{xB}[t] \\
f_{yB}[t] \\
f_{zB}[t] \\
-z_{inB} f_{yB}[t] + y_{inB} f_{zB}[t] + m_{xB}[t] \\
z_{inB} f_{xB}[t] - x_{inB} f_{zB}[t] + m_{yB}[t] \\
-y_{inB} f_{xB}[t] + x_{inB} f_{yB}[t] + m_{zB}[t]
\end{pmatrix}
$$

(4.13)

4.3 THE LAGRANGE'S EQUATIONS OF MOTION

As already mentioned, the Lagrange's equation is concisely coded in *Mathematica* as

```
LagrEqns[T_, U_, Damp_, Q_List, Coor_List] :=
 Flatten[MapThread[{∂t(∂∂t#2 T) - ∂#2 (T - U) + ∂∂t#2 Damp - #1 == 0} &,
  {Q, Coor}]];
```
(4.14)

With all energies (potential, kinetic, and dissipative) defined, all input force and moment, and coordinates of the platform and base, this "LagrEqns" function generates twelve equations of motion. All these equations involve symbolic variables. In the remaining parts of this thesis, a simplified model is analyzed. It is obtained by assuming a fixed base position. Thus, the number of equations of motion reduces to six. Similarly, six transfer functions represent the dynamic characteristics of the dynamometer's platform.

4.4 BODE PLOT FROM TRANSFER FUNCTION

By using *Mathematica* 'Control System Professional', representative transfer function, are obtained and their Bode plots are shown in Fig.4.4. Resonance frequencies of the system can are computed and shown in Table 4.1. Bode plots and resonance frequencies are in good agreements with experimental results processed in Chapter 5.

Transfer Function	Natural Freguency
G_{xx}	$\cong$ 600 Rad/Sec
G_{yy}	$\cong$ 600 Rad/Sec
G_{zz}	$\cong$ 1250 Rad/Sec

$G_{i,j}$: i direction response in j direction input.

Table 4.1 Estimated Natural Frequencies from Bode Plots of TF.
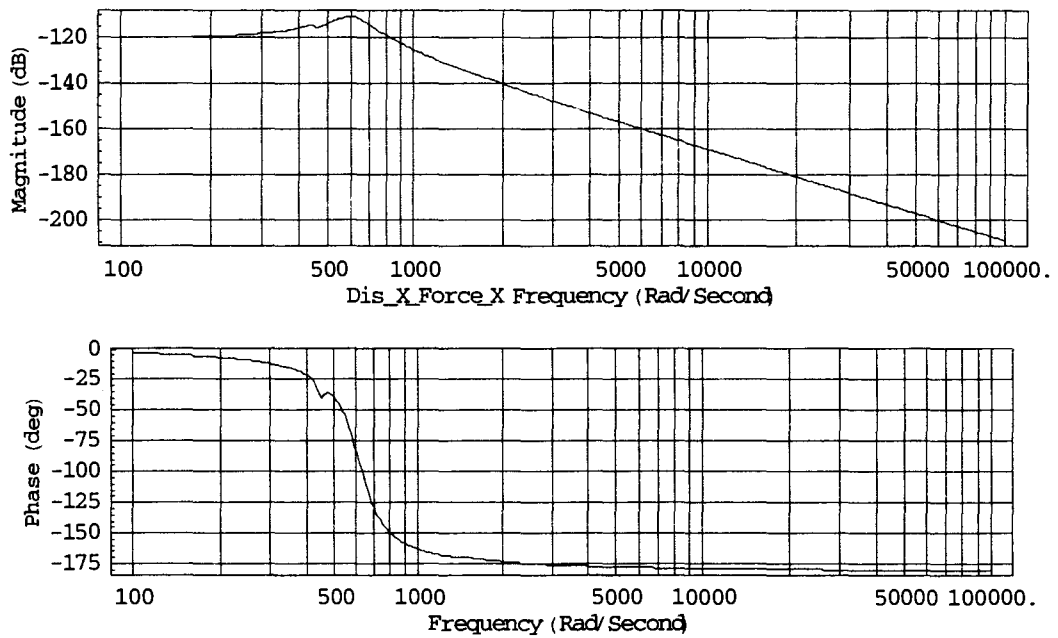
Figure 4.4a Bode plot for transfer function between the force and displacement in the x direction.
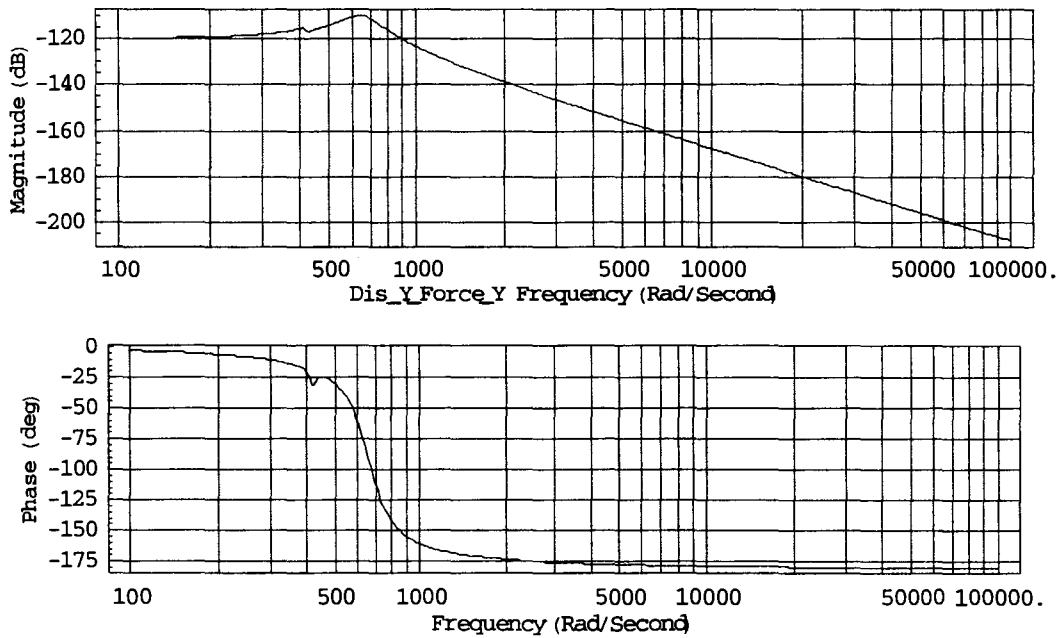
Figure 4.4b Bode plot for transfer function between the force and displacement in the y direction.
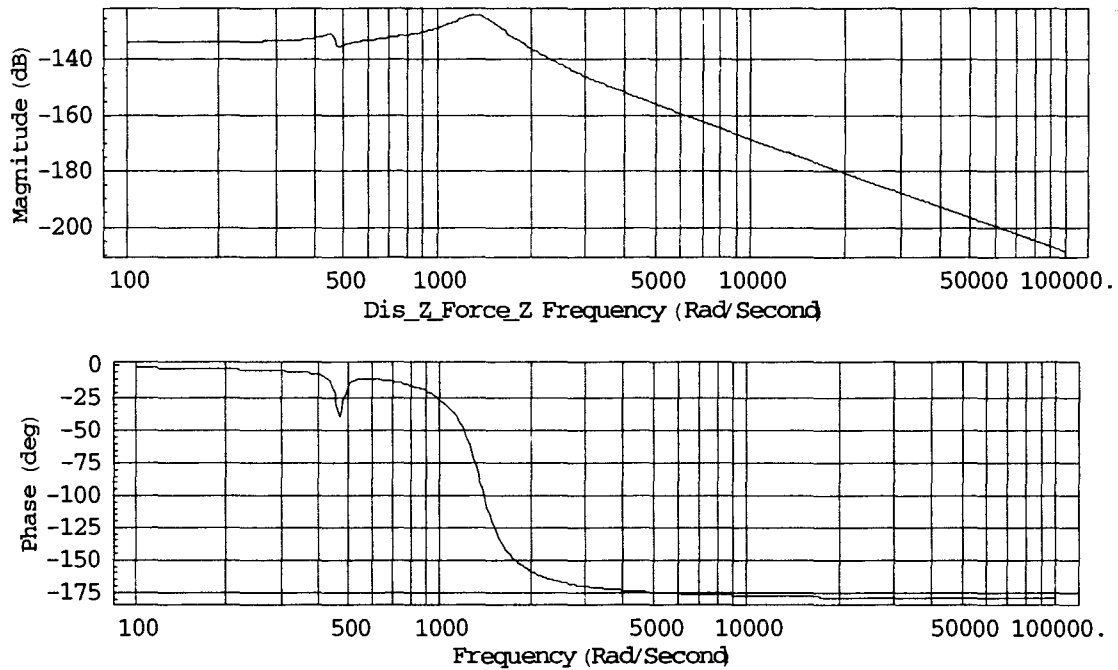
Figure 4.4c Bode plot for transfer function between the force and displacement in the z direction.

4.5   SIMULATION OF MODEL RESPONSE TO ACTUAL IMPACT

The transfer functions discussed in the previous section allow obtaining the response of the model to experimental signals. Since these transfer functions are continuous, and it is necessary to convert them into discrete form required for the compatibility with discretized experimental records. There are several discretization techniques that can be used for simulating continuous-time system. The most important of them are shown below [Katsuhiko Ogata 1987].

1. Backward difference method.

2. Forward difference method. Since this method may lead to an unstable, it should be used with cautions.

3. Bilinear transformation method (a numerical integration method base on the trapezoidal integration rule).

4. Impulse-Invariance method (impulse-invariance method with sample-and-hold – the z transform based method coupled with a fictitious sample-and-hold).

5. Matched pole-zero mapping method.

These different methods yield slightly different discrete-time systems. The bilinear transformation method has been chosen in this research.

4.5.1 Bilinear transformation method

Fig. 4.5 shows the area approximation by the bilinear transformation method. y[k] is representing the left part area from the time kT, and y[k-1] is representing the left part area from the time $kT - T$.
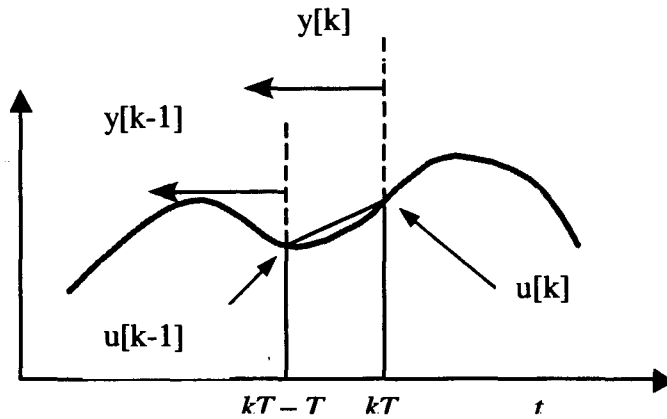
Figure 4.5 Trapezoidal integration.

Considered is a transfer function D(s) of a system with the input U(s) and the output Y(s).

$$\frac{Y(s)}{U(s)} = D(s) = \frac{1}{s}, \tag{4.15}$$

In the discrete time, this transfer function represents the following integral equations.

$$y(kT) = \int_0^{kT-T} u(t)dt + \int_{kT-T}^{kT} u(t)dt \qquad (4.16)$$

This equation can be next approximated by a recursive discrete time equation

$$y(k) = y(k-1) + \frac{T}{2}[u(k-1) + u(k)] \qquad (4.17)$$

Applying the z-transformation to the above equation yields

$$\frac{Y(z)}{U(z)} = \frac{T}{2}\left(\frac{1+z^{-1}}{1-z^{-1}}\right) \qquad (4.18)$$

Comparison of Eq.4.15 and Eq.4.18 gives the relationship between the s-domain and the z-domain transfer function. It suggests that by substituting

$$s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right) \qquad (4.19)$$

in the continuous time transfer functions, respective discrete time D(z), the transfer functions can be obtained.

This transformation method, referred to as Tustin's or the bilinear [Franklin, 1994], is available in 'Control System Professional' package, for rapid symbolic conversion from the continuous-time to the discrete-time domains. The following function needs to be called.

```
myDiscrTF = ToDiscreteTime[myTF, Method→ BilinearTransform,
   CriticalFrequency→ Automatic, Sampled→ Period[ts]]
```
$$\qquad (4.20)$$

4.5.2 Impulse-invariance method

By using the inverse-Laplace transformation and z transformation, the equivalent discrete-time transfer function $G_D(z)$ can be obtained as follows.

$$G_D(z) = \mathcal{Z}[g_D(kT)] = T\mathcal{Z}[g(t)] = T\mathcal{Z}[\mathcal{L}^{-1}[G(s)]] = TG(z) \qquad (4.23)$$

Where, the inverse z transformation of $G_D(z)$ is $g_D(kT)$, the discrete time transfer function, and this is T times of $g(t)$. This $g(t)$ is also expressed as $\mathcal{L}^{-1}[G(s)]$.

As a example suppose a continuous-time system is described by a transfer function as

$$G(s) = \frac{a}{s+a} \qquad (4.22)$$

Then, the equivalent discrete-time transfer function is as below

$$G_D(z) = TG(z) = \frac{Ta}{1-e^{-aT}z^{-1}} \qquad (4.24)$$

Since $G_D(z)$ is proportional to the z transform of the continuous-time transfer function, so the impulse–invariance method is also called the "z transform method" [Katsuhiko Ogata 1987].

4.5.3 Simulation of response to the actual force impact

By using the actual impact signal from an instrumented hammer together with the discrete-time transfer-function, derived for the model under investigation, the realistic impulse response of modeled system can be simulated. This response is readily obtained in *Mathematica* by executing the following command.

```
myResp = OutputResponse[myDiscrTF, Take[impactVectorCalibr, {1, dataSize}]];
```

$$(4.21)$$

In which

      "myResp" : simulated response,

      "myDiscrTF" : discrete transfer function obtained from Eq.4.20, and

      "ImpactVectorCalibr : the experimentally recorded impact force.

(a)   (b)
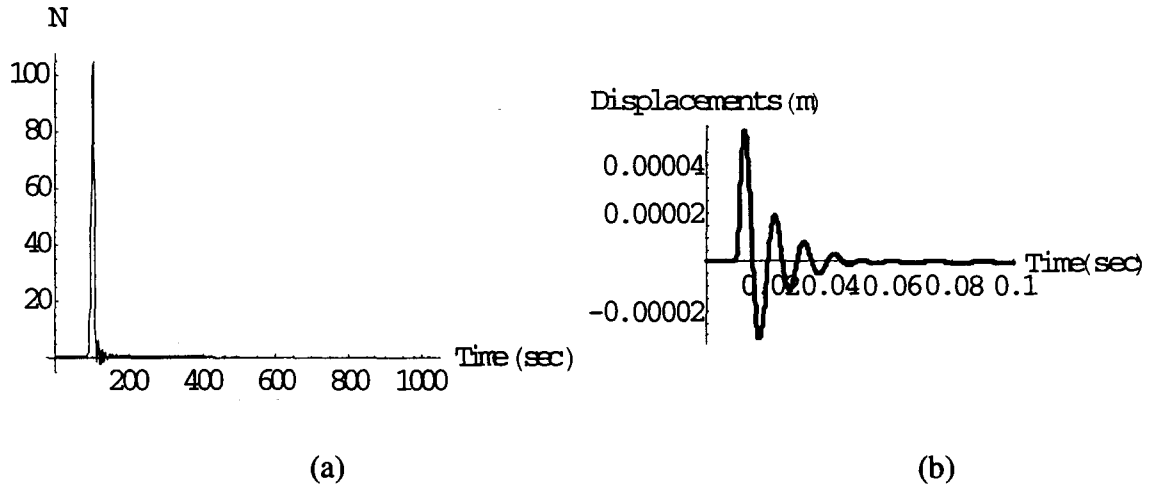
Figure 4.6 (a) Calibrated impact force, (b) simulated impact response of the system in the y direction(obtained from the transfer function $G_{yy}(s)$).

4.6 CLOSURE

By applying the Lagrange's energy formalism and symbolic method such as 'Mathematica', the equations of motion are readily derived. From the Bode plots of the transfer functions ($G_{i,j}$) in the x, y, and, z directions natural frequencies are estimated. These frequencies characterize the model-based responses of the dynamometer under consideration. In the next chapter, they are compared with signal-based responses to validate the presented modeling methodology.

5. EXPERIMENTAL VALIDATION

In previous chapter, the equations of motion of dynamometer are derived. Under the assumption of no foundation and base displacement, simplified transfer functions are obtained, and these transfer functions provide agreeable bode plots which show reasonable natural frequencies for the model under consideration. For the next steps, the validation is taken by the use of experimental test. As a response of an impact force on the platform of dynamometer, the movement of the platform is calculated by using several experimental techniques. These techniques include data acquisition procedure, numerical double integration, and several processes for eliminating drift. Simulated responses that come from the model of dynamometer will be compared with the results that are captured and calculated from the experimental test.

5.1 DATA ACQUISITION SYSTEM

A standard data acquisition (DAQ) system comprises the following basic components:

(1) a controller, (2) a signal conditioner, (3) a multiplexer and amplifier, (4) an analog-to-digital converter (ADC), (5) a storage unit or a memory unit, and (6) a readout device [Dally et al.,1993]. In the DAQ system used in this research, data from the sensors (accelerometers) is stored in desktop computer equipped with DAQ board. A LabVIEW® program is used for controlling the DAQ system. The ADC, multiplexer and amplifier are provided in a plug-in DAQ printed circuit board type AT-MIO16E2 from National Instruments [1994]. Low-pass filters serve as signal conditioners to prevent signal aliasing. A schematic diagram of the employed DAQ system is shown in Fig. 5.1 [Jitpraphai, 1997].
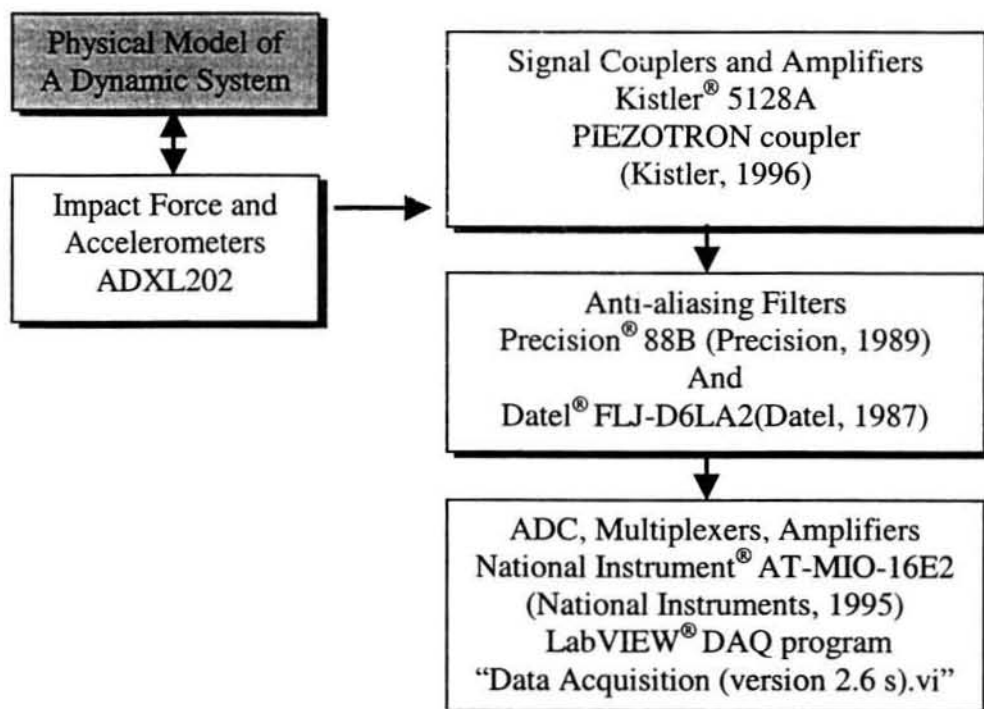
Figure 5.1 Block diagram showing the data acquisition system used in this research.

5.1.1 Overview of the Methodology

The acceleration signals from each sensor are recorded by a data acquisition (DAQ) system. The DAQ is controlled by a LabVIEW®-based *Data Acquisition Controller program*, developed in previous research [Jitpraphai, 1997]. To obtain displacements of the dynamometer's platform from the voltage signals generated by accelerometers placed on the platform, the following steps are needed;

1) Amplifying the signals.
2) Filtering the signals.
3) Converting the voltage signals to acceleration signals.
4) Double integration to obtain a 'rough' estimate of displacements.
5) Eliminating the drifts to obtain accurate estimates of displacements

5.1.2 Data Acquisition Program

A controller program for data acquisition (DAQ) is required to read data from accelerometers. The program "DAQ Controller.VI" written in the LabVIEW®'s G language is used in this research [Jitpraphai, 1997]. This is an interface program between the user and a DAQ board AT-MIO 16E2 [National Instrument, 1995]. The user can command the board to acquire analog voltage signals with desired parameters. The user can readily inspect the acquired signals, select suitable sampling parameters for these signals and acquire data again with optimized parameters.

5.2 EXPERIMENTAL SETUP

By using impact hammer, transient excitation is applied to the experimental model of a Kistler® dynamometer type 9257A [Kistler, 1996]. Several acceleration signals (the system's responses representing vibrations) were recorded and processed to obtain the movements of the dynamometer's platform. Instruments are set up according to the schematic diagram shown in Fig. 5.2.
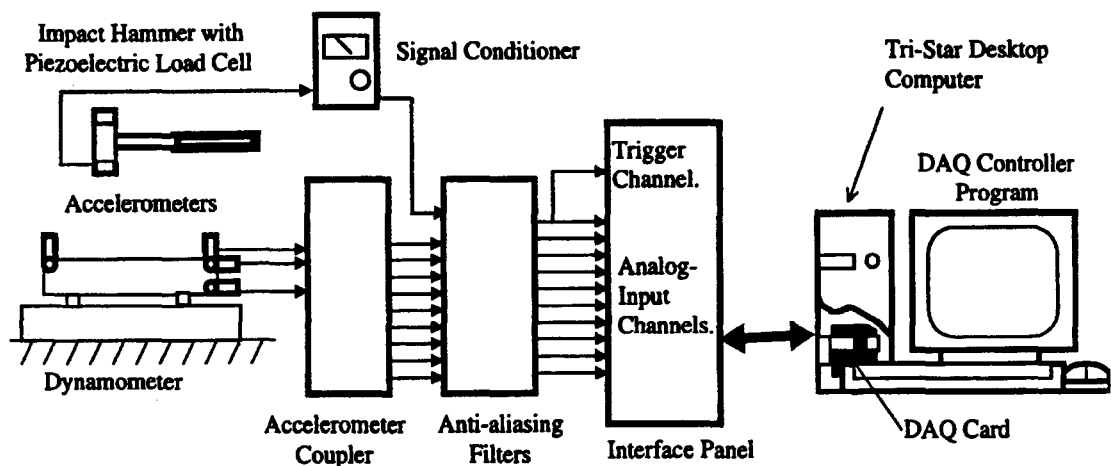
Figure 5.2 Schematic diagram of the experimental setup.

5.3   SIGNAL PROCESSING IN EXPERIMENTAL DATA

Signal processing procedure includes several steps required for the calculation of the actual displacements from the 'raw' voltage signal generated by the accelerometers. The signal processing steps are summarized in Fig. 5.3.

5.3.1   Conversion to Physical Units and modification

Raw voltage data that is acquired from an accelerometer needs several processing steps to be converted to displacement. First, this data should be converted to physical units of acceleration i.e. m/s^2. The calibration equation is

$$a_{cal}[m/s^2] = 9.81 \left[\frac{m}{s^2 \cdot g} \right] \cdot \frac{a_s}{c_s} \qquad (5.1)$$

Where, a_{cal} - calculated acceleration.

a_s - acceleration signal from sensor [V].

c_s - calibrated factor [V/g] (sensitivity).

Sensitivity is the parameter of accelerometer that is specified by its manufacturer. Two kinds of accelerometers are used in this research and the sensitivities of these sensors are given in Appendix A.1 (ADX202 : 0.312 V/g, Kistler : 0.5 V/g). The signals, from ADX202 accelerometers are pre-amplified 5 times by their respective hardware circuitry. This gain should be considered in the conversion defined in Eq. 5.1. After calibration, data is ready for the integration. Ideally, the data from accelerometer should be centered, in other words the acceleration before the impact and after the transient part should be zero. Realistically, it is not the case. Non-centered data can cause large errors of numerical integration. To avoid these errors, subtraction of the average signal value before the integration is required. Another centering of data is required before the second integration. Finally displacement is obtained, but it is still severely distorted. Even in the data obtained from high

performance sensors, the distorted signal is useless without additional processing. Eliminating distortion without affecting the shape of the actual measured displacement is required. There are several methods to accomplish this, and these methods are discussed in Section 5.3.3.


```
┌─────────────────────────────────────────────┐
│     Voltage Signals from Accelerometers      │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│ Conversion to Physical Units of Acceleration, i.e., m/s² │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│              Acceleration Signal             │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│   Subtracting the Average or Moving-average Values   │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│           First Numerical Integration        │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│               Velocity Signals               │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│         Subtracting the Average Velocity     │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│          Second Numerical Integration        │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│               Eliminating drifts             │
└─────────────────────────────────────────────┘
                      ↓
┌─────────────────────────────────────────────┐
│             Displacement Signals             │
└─────────────────────────────────────────────┘
```
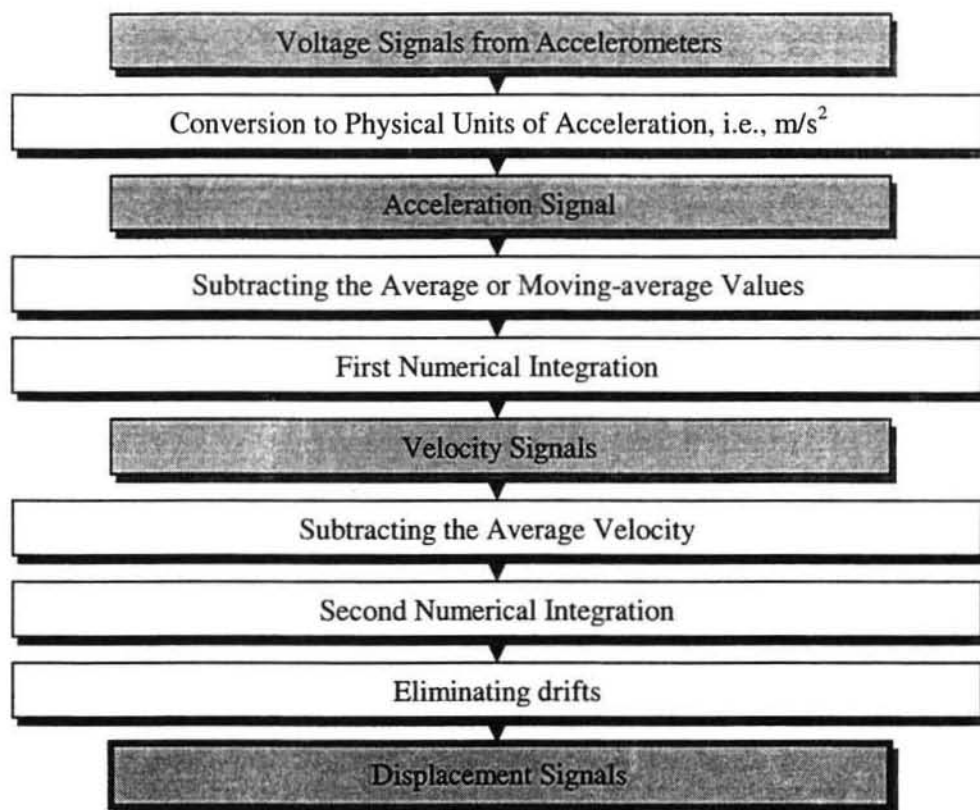
Note: Clear blocks represent operations while the shadowed blocks represent signals.

Figure 5.3 Flowchart of signal processing.

5.3.2   Double Integration Procedure

After modifying the raw acceleration signal, a rectangular rule[6] numerical
integration method [Yakowitz, 1989], is performed by using *Mathematica* to
calculate velocities. This is accomplished by the following code:

```
f[0] = 0;
f[n_] := f[n] = f[n- 1] + accel1SI[n] ts;
velocity = Table[f[n], {n, 1, dataSize}];
```
                                                                           (5.2)

Where accel1SI - an array of recorded and pre-processed data.
          velocity - an array of integrated acceleration data.

An example velocity calculated by numerical integration of the calibrated and
pre-processed acceleration is shown in Fig. 5.4. Before performing the second
integration of this velocity signal, zero centering by subtracting the average value is
applied again, and second integration is performed as

```
velocity = Table[f[n], {n, 1, dataSize}];
displacement = Table[h[n], {n, 1, dataSize}];
```
                                                                           (5.3)

The obtained displacement signal is shown in Fig.5.5.
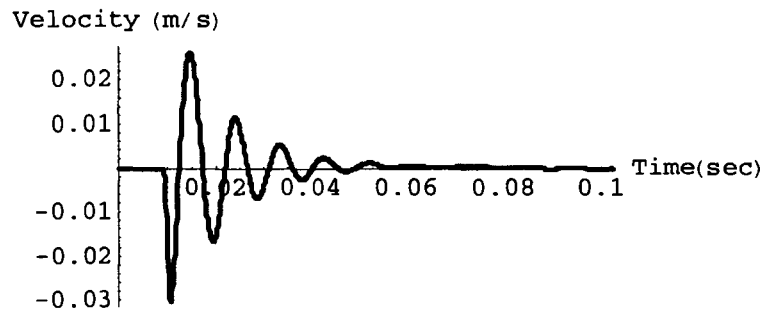
Figure 5.4 Velocity calculated by numerical integration of the calibrated
acceleration signal.

---

[6] Acceptable due to high sampling frequency
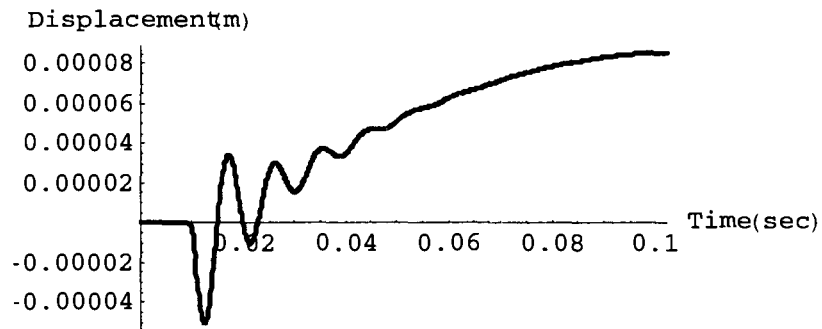
Figure 5.5 Displacement calculated by numerical double integration.

For comparison, a response simulated by means of the derived analytical model is shown together with displacement obtained by double integration in Fig. 5.6.
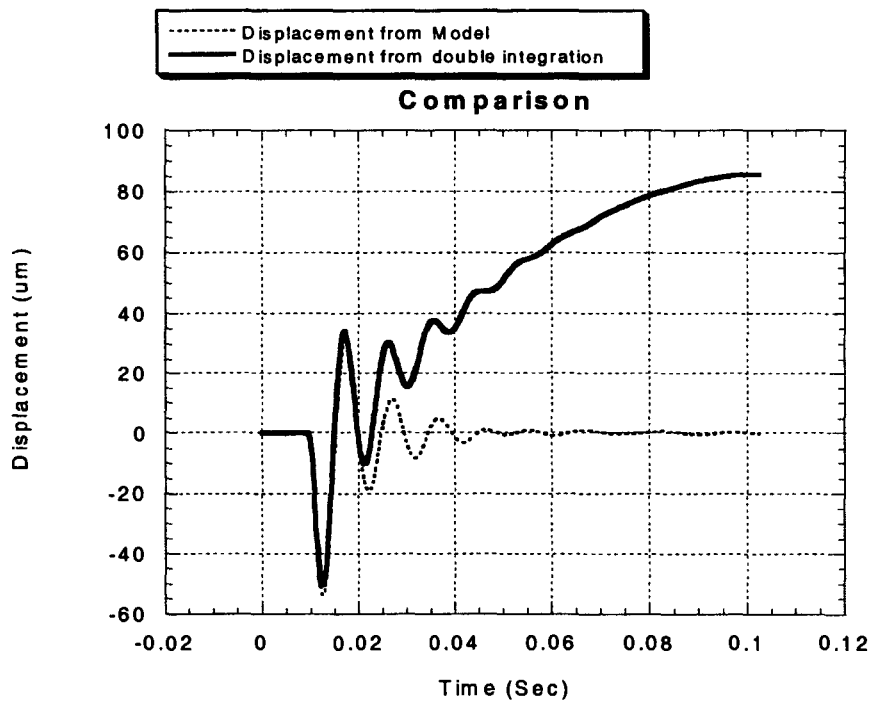
Figure 5.6 Displacement obtained from the analytical model and by double integration of the experimental acceleration occurred from impact.

5.4   COMPARISON FOR VALIDATION

From twelve equations of motion derived by Lagrange's equation in Chapter 4, twelve transfer functions are obtained. As already mentioned, for the sake of brevity, the above model was simplified by assuming fixed base, and for the each model, six transfer functions have been obtained. Responses to an impact force in every coordinates of the platform was simulated from these transfer functions. In the remaining part of this chapter, these simulated responses will be compared with the responses obtained by the method described above. In addition, various methods of estimating the "true" drift in double integrated acceleration signal will be considered. The accelerometers are mounted on the faces of platform as shown in Fig. 5.8. Three acceleration signals need for the translational movements, and six more signals are required to obtain the rotational movements without using numerical solution [Padgaonkar, 1975; Lie, 1976]. By using two direction signals capturing accelerometer such as "ADXL202", with 5 sensors, 9 required signals accelerations can be obtained. More details are discussed in Chapter 6.

As discussed in section 4.5.1, by using 'Bilinear transformation' method, transfer functions are described in discrete time domain, and with these, the impact responses of the system are simulated. The displacements obtained from this simulation can be verified by direct comparison with the displacements that are obtained from double integration and simple procedure of eliminating drift. Fig. 5.8, and Fig. 5.9 are showing the figures of the experimental test, and the several strong displacements, which are calculated from the test. Fig. 5.8 is showing the displacements in same direction of the applied impact force. These responses are strong and close to the simulated responses as shown in Fig. 5.7. In the other hand, the displacements shown in Fig. 5.9 are weak responses, and even after the elimination of drift by simple signal processing procedure, there still remained distortions that should be rectified. In Chapter 5.5, several methods are discussed for reducing the error.
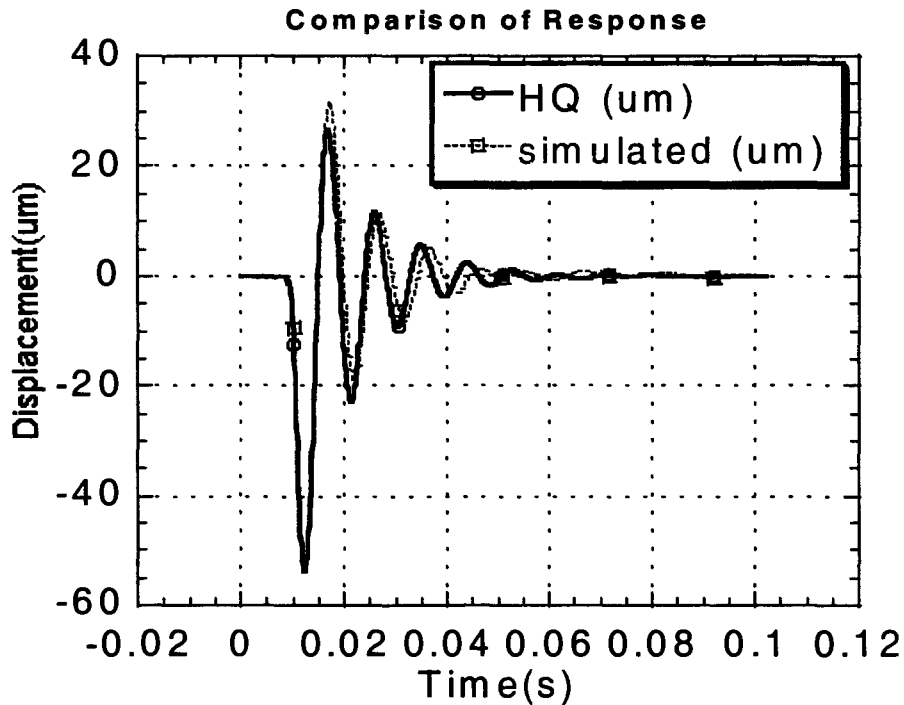
Figure 5.7 Displacement comparison between the model and experimental test.

5.5 MINIMIZING ERRORS

Clearly, there is a significant drift in the displacement as shown in Fig. 5.6 that should be eliminated. To rectify the current data that includes the drift, several models that represent the drift are introduced. For example, by using selective exponential model or a polynomial model, the drift part of the displacement can be represented. Subtracting these models from the distorted displacement yields the actual displacement as shown in Fig. 5.10.

Fig. 5.10.b shows a realistically looking displacement of the dynamometer under consideration. But in many cases, especially for the weak signals, just eliminating sensor drift by the above-presented simple model does not give satisfactory results. So several improved methods for eliminating drift are discussed below.
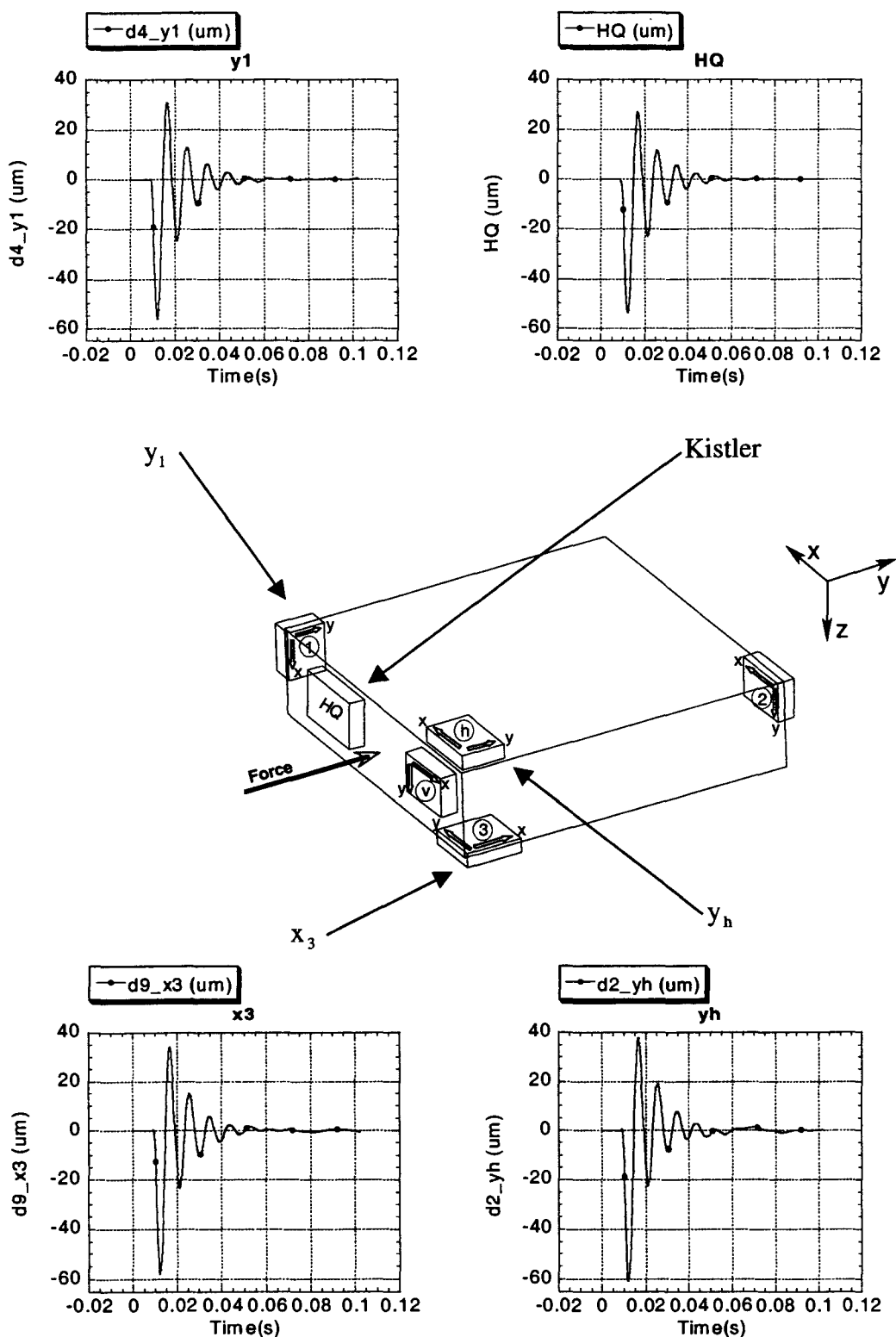
Figure 5.8 Location of nine sensors on the platform and example strong signals obtained from the experiment.
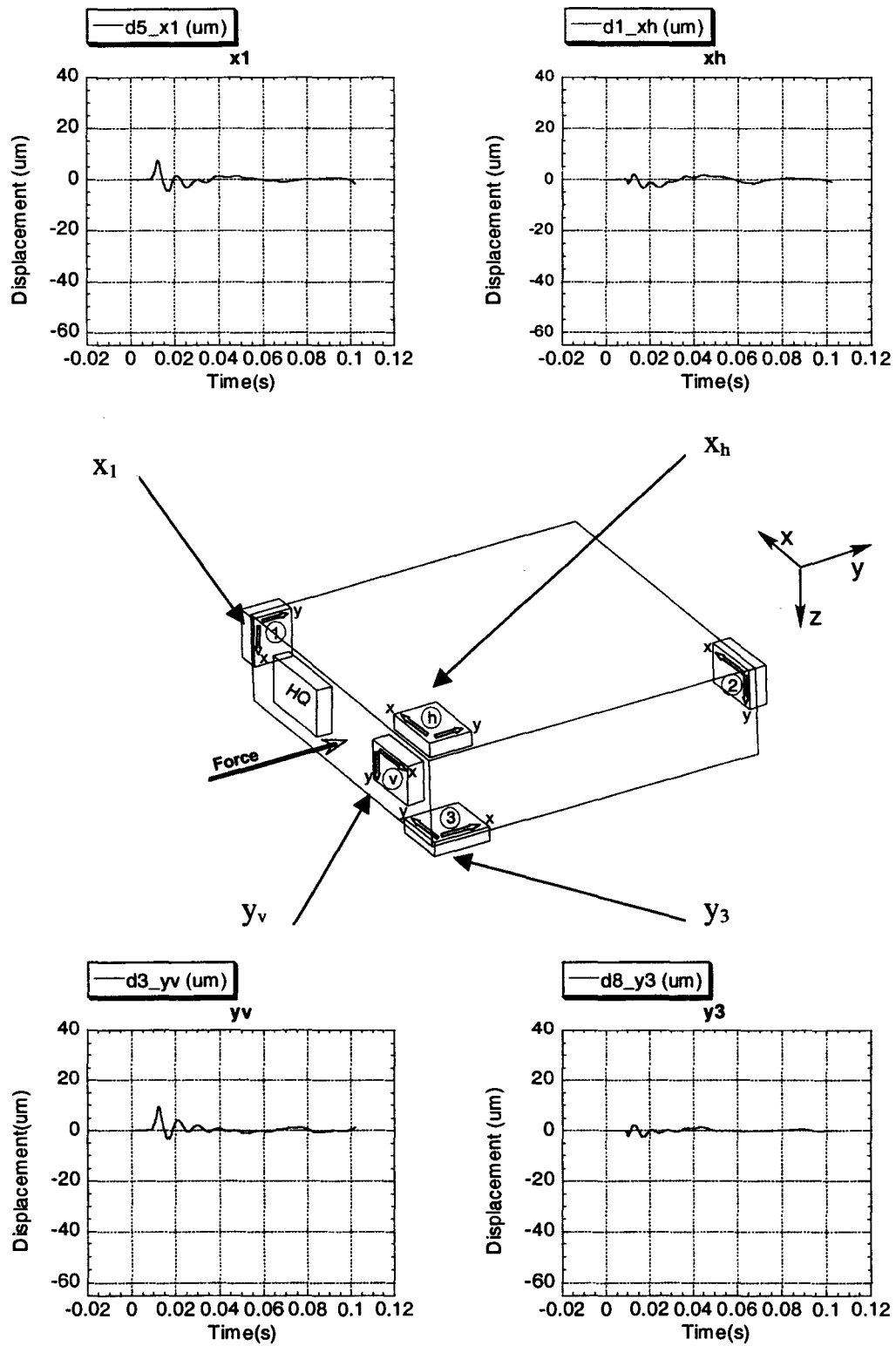
Figure 5.9 Location of nine sensors on the platform and example weak signals

obtained from the experiment.

a.   b.
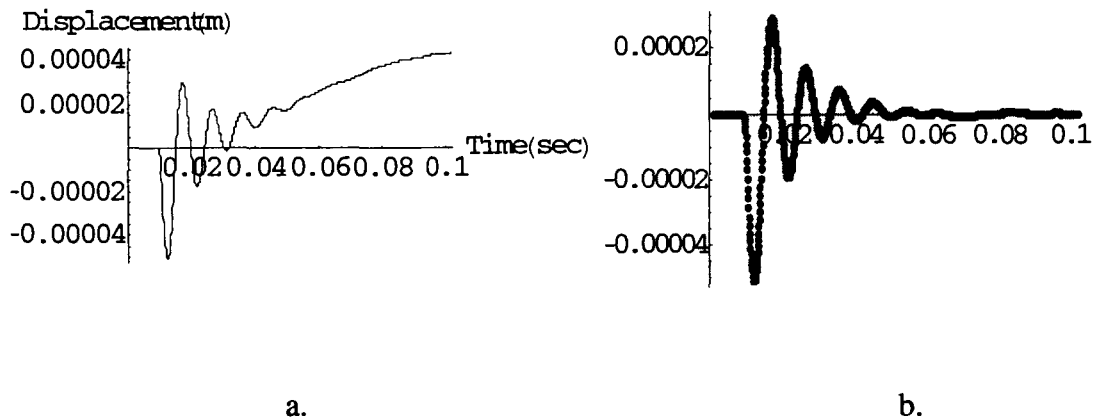
Figure 5.10 Double integrated acceleration signal used for polynomial estimation (a), and displacement after drift elimination (b).

5.5.1   Eliminating Drift by Piecewise Polynomial Model

This is the simplest way to eliminate drift. Considering the double integrated acceleration signal with drift as shown in Fig. 5.11, by excluding the "transient" part of the signal, two separate sections shown in Fig. 5.12 with thick lines are obtained. This curve can be well fitted with a polynomial. Assuming this fitted curve represents the drift, the actual displacement is obtained by subtracting this fitted curve from the original curve. The result is shown in Fig.5.13.
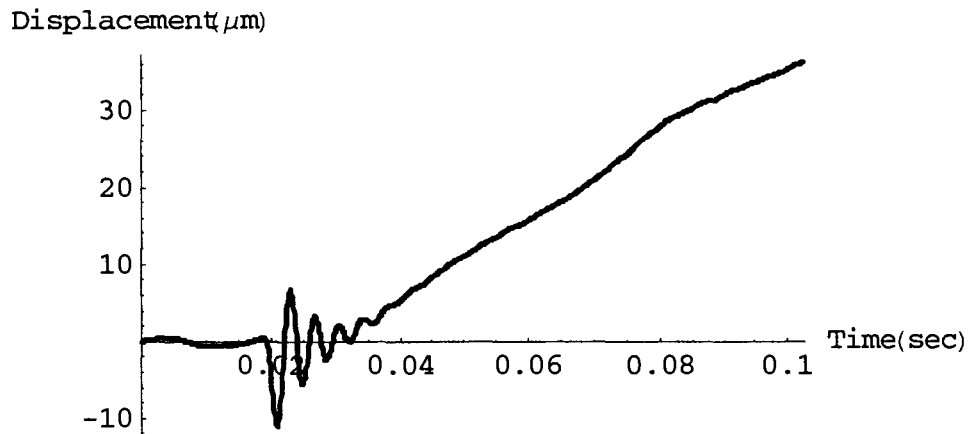
Figure 5.11 Displacement with drift
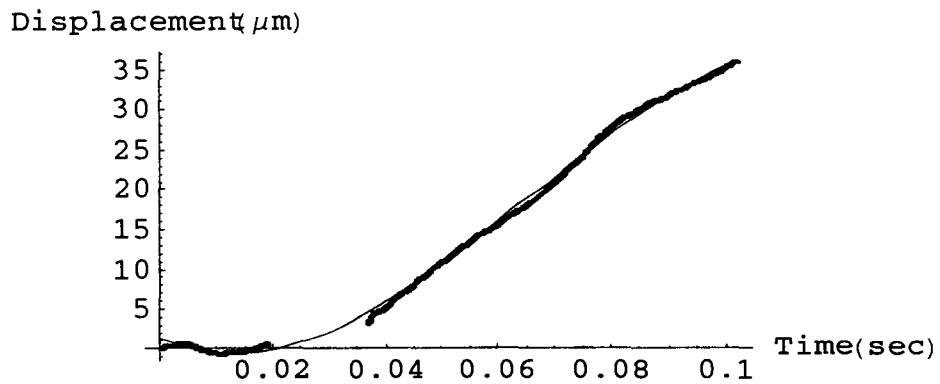
Thick lines: data to be fitted. Thin line: fitted polynomial.

Figure 5.12 Piecewise Displacement estimation with Polynomial model.
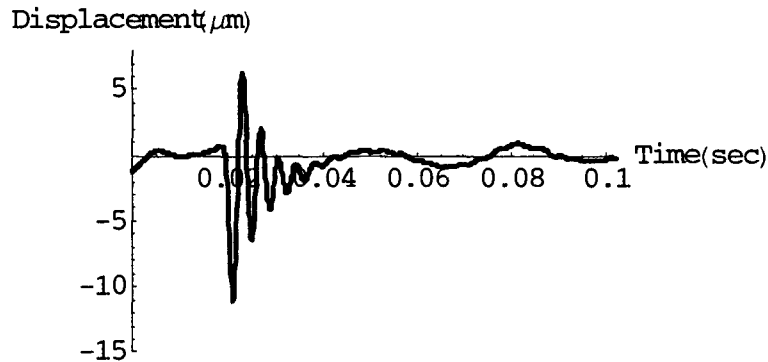
Displacement(μm)

Figure 5.13 The estimated displacement obtained after eliminating drift.

Although attractive for its simplicity and robustness, this method rarely gives satisfactory results. Even changing the order of the polynomial model to 5th does not improve the result much. As can be seen in Fig. 5.13, the displacement exhibits oscillations, which in this case cannot be attributed to the dynamics of the tested dynamometer. Therefore more accurate curve fitting procedure should be considered. One such procedure is obtained by performing two separate fits on suitably chosen data subsections as shown in Fig. 5.14. A small improvement is made in beginning part, but still the result is not satisfactory as shown in Fig.5.15.
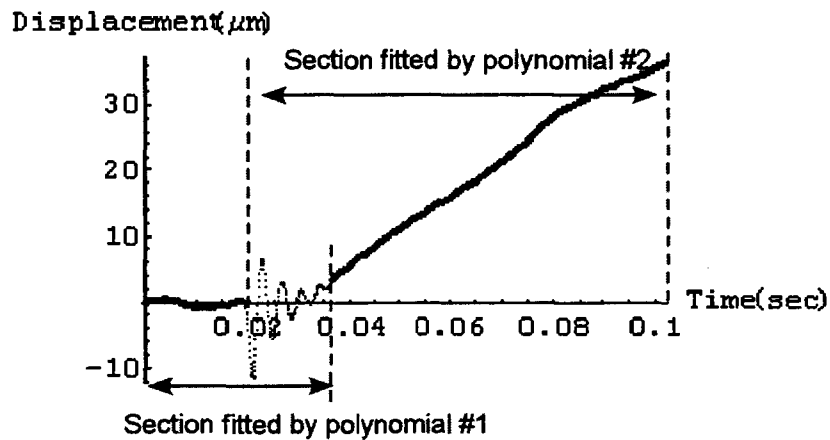
Figure 5.14 Separated curves for individual fitting.
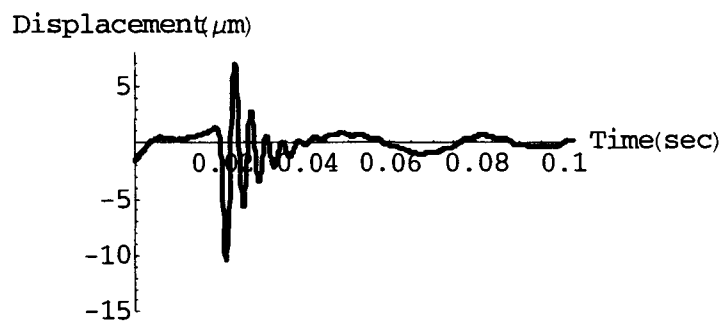
Figure 5.15 Displacement after drift elimination

5.5.2 Eliminating Drift by Using Polynomial & Impact Response Models

By using a polynomial model that will represent drift, and an impact response model that will represent the displacement response of system, better displacement

estimation can be obtained. If the sum of response of these models (polynomial and impact response) can fit satisfactorily the double integrated acceleration siganl, the impact response model can be considered as a representative for the displacement. Nonlinear fitting code in '*Mathematica*' is shown in Eq. 5.5. After defining polynomial and impact response model with symbolic coefficients, by using the 'NonlinearRegress' function, the coefficients of the function that provide best fit of curve for the double integrated acceleration signal are readily determined. The equation of the function used for fitting is as below.

$$f_{fit}(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + a_5 x^5 + A \cdot Exp(x - x_0) \cdot Sin(b(x - x_0) + c)$$

(5.4)

in Mathematica code, it is as below.

```
lclRes = NonlinearRegress[dataForPlotting,
    a0 + a1 x + a2 x² + a3 x³ + a4 x⁴ + a5 x⁵ +
    a6 Exp[ - a7 (x - delay)] * Sin[a8 (x - delay) + a9] * UnitStep[x - delay], x,
    {a0, a1 , a2 , a3 , a4, a5 , a6, a7, a8, a9}, MaxIterations -> 300,
    RegressionReport -> BestFitParameters]
```

(5.5)

Due to strong non-linearity and multiple minima of the minimized cost function, this simple procedure in general is not satisfactory since it does not converge to the global minima.

5.5.3 Polynomial & Impact Response Model with User Specified Initial Values

By using the function 'FindMinimum' in *Mathematica*, the coefficients of the model that fit the original curve most closely can be obtained. One initial value per an estimated coefficient that is close to the global minimum is required by this method. Note that the procedure presented in Section 5.3.2 did not require specifying the initial conditions.

The function used to fit the double integrated acceleration is a sum of 5th order polynomial and time lifted impact response function of the 2nd order system. This function is defined with the name of "func1" as shown below.

func1[a0_, a1_, a2_, a3_, a4_, a5_, a6_, a7_, a8_, a9_, x_] :=
a0 + a1 x + a2 x^2 + a3 x^3 + a4 x^4 + a5 x^5 +
a6 Exp[- a7 (x - delay)] * Sin[a8 (x - delay) + a9] UnitStep[(x - delay)]   (5.6)

The function "residual" is defined by subtracting the defined function, "func1" from the raw displacement (displacement with drift). The square sum of "residual" is defined as "chisqr" as shown in Eq.5.8, and is used for finding minimum as shown below.

$$J(a_0, a_1, ..., a_9) = \sum_{i=1}^{N} \left[f_d(i) - r(x_i; a_0, a_1, ... a_9) \right]^2 \qquad (5.7)$$

$J(a_0, a_1, ..., a_9)$: square sum of 'residual',

$f_d(i)$: displacement in time of $\dfrac{i}{10000}$ sec.,

$r(x_i; a_0, a_1, ... a_9)$: function used to fit the double integrated acceleration signal.

.

residual[a0_, a1_, a2_, a3_, a4_, a5_, a6_, a7_, a8_, a9_, i_] :=
(func1[a0, a1, a2, a3, a4, a5, a6, a7, a8, a9, i / 10000] - displacement[i])

chisqr[a0_, a1_, a2_, a3_, a4_, a5_, a6_, a7_, a8_, a9_, n_] :=
Sum[(residual[a0, a1, a2, a3, a4, a5, a6, a7, a8, a9, index])^2,
{index, 1, n}]   (5.8)

Once the performance index "chisqr" is defined, it is minimized in *mathematica* by the command of below.

```
res = FindMinimum[chisqr[a0, a1, a2, a3, a4, a5, a6, a7, a8, a9, (1000)],
   {a0, -0.04}, {a1, -0.031}, {a2, 0.068}, {a3, -0.025}, {a4, 0.004},
   {a5, -0.00021}, {a6, -1.4}, {a7, 149.1}, {a8, 1550.8}, {a9, 2.387},
   MaxIterations -> 30]
```
(5.9)

Initial values of the coefficients of the fitted model are provided in the command. This method suffers from similar weakness as the simpler function in Section 5.3.2, namely, it tends to converge to the nearest local minimum. Consequently the obtained coefficients do not provide satisfactory model for the drift and response of the system. This disadvantage is exacerbated by using gradient minimization technique. Finding minimum with bracketing initial values is recommended.

5.5.4 Polynomial & Impact Response Model with Bracketing Initial Values

There are two methods for minimization technique, namely gradient method and non-gradient method. Gradient method requires computation of gradients. Obviously, if an objective function is not differentiable, or if it is very difficult to compute its gradient, this method cannot be used. And there is no guarantee that a solution returned by gradients method is a global minimum [Bhatti, 1998]. On the other hand, non-gradient method requires only the function values or generates an approximation of a gradient vector by using finite differences. This method takes more time and expenses, but there are several virtues of non-gradient method that make them worth the price. In general, non-gradient method tends to be more reliable than gradient method, and provides a means for solving a problem which is insoluble in gradient method [Hansen, 1992]. By the option of using bracketing initial values in "FindMinimum', non-gradient method can be applied in *Mathematica*. By using the bracketing initial values, the likelihood of finding the global minimum is increased. And as expected, this method takes more time

depends on the size of bracket's range. *Mathematica* code for finding minimum by using bracketing initial values is written in Eq. 5.13.

$$
\begin{aligned}
&\texttt{resD = FindMinimum[chisqr[a0, a1, a2, a3, a4, a5, a6, 1024],}\\
&\quad \texttt{\{a0, \{0.2 a0}_{\texttt{init}}\texttt{, 4 a0}_{\texttt{init}}\texttt{\}\}, \{a1, \{4 a1}_{\texttt{init}}\texttt{, 0.2 a1}_{\texttt{init}}\texttt{\}\},}\\
&\quad \texttt{\{a2, \{0.2 a2}_{\texttt{init}}\texttt{, 4 a2}_{\texttt{init}}\texttt{\}\}, \{a3, \{4 a3}_{\texttt{init}}\texttt{, 0.2 a3}_{\texttt{init}}\texttt{\}\},}\\
&\quad \texttt{\{a4, \{4 a4}_{\texttt{init}}\texttt{, 0.2 a4}_{\texttt{init}}\texttt{\}\}, \{a5, \{0.2 a5}_{\texttt{init}}\texttt{, 4 a5}_{\texttt{init}}\texttt{\}\},}\\
&\quad \texttt{\{a6, \{0.2 a6}_{\texttt{init}}\texttt{, 4 a6}_{\texttt{init}}\texttt{\}\}]}
\end{aligned}
\tag{5.10}
$$

Where, the function for fitting is defined as below.

$$
\begin{aligned}
&\texttt{func2[a0\_, a1\_, a2\_, a3\_, a4\_, a4\_, a5\_, a6\_] :=}\\
&\texttt{a0 + a1 x + a2 x}^2 \texttt{ + a3 x}^3 \texttt{ + a4 Exp[-a5 x] Sin[a6 (x - 0.0197)] UnitStep[x - 0.0197]}
\end{aligned}
$$

(a)   (b)
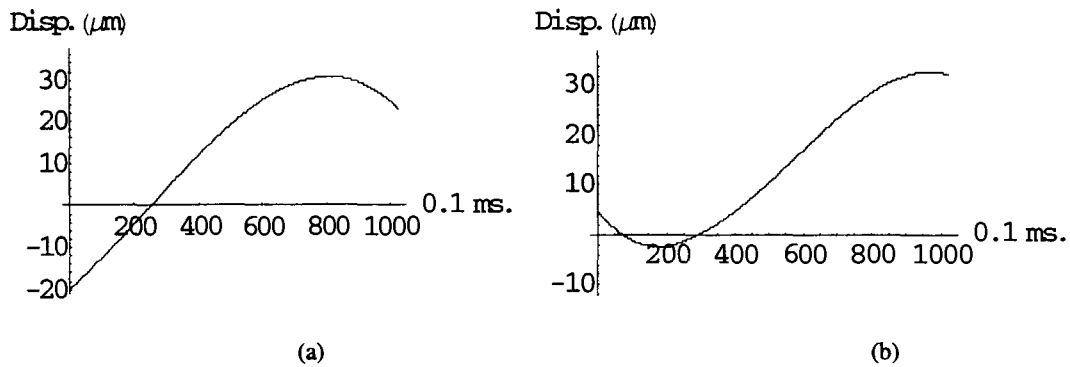
Figure 5.16 Comparison of minimization between gradient initial method (a), and bracketing initial values method (b).

Fig. 5.16 shows an example of the difference between using a gradient search method and using the quadratic minimization method. The user specified initial values were bad for both case, but as shown in Fig.5.16, the result has big difference. Even with bad initial values, brackets method gives better curve fitting than the method of using gradients. But it is still not satisfactory unless the specified initial values are well predicted.

5.5.5 Polynomial & Impact response Model by using "MultiStartMin"

By using "MultiStartMin[7]", which is special package in *Mathematica*, satisfactory curve fitting is obtained as shown in Fig.5.17. By eliminating the polynomial model from the fitted curve, the fitted impact response is obtained as shown in Fig.5.18.

5.6 CLOSURE

Rigorous numerical comparison of displacements simulated from the analytical model and the experimental test shows a good agreement. This validates the assumption underlying the model development. In next chapter, by using visualization, the simultaneous comparison of three-dimensional motion will be performed.

Black line: fitted curve; Gray line: displacement with drift.
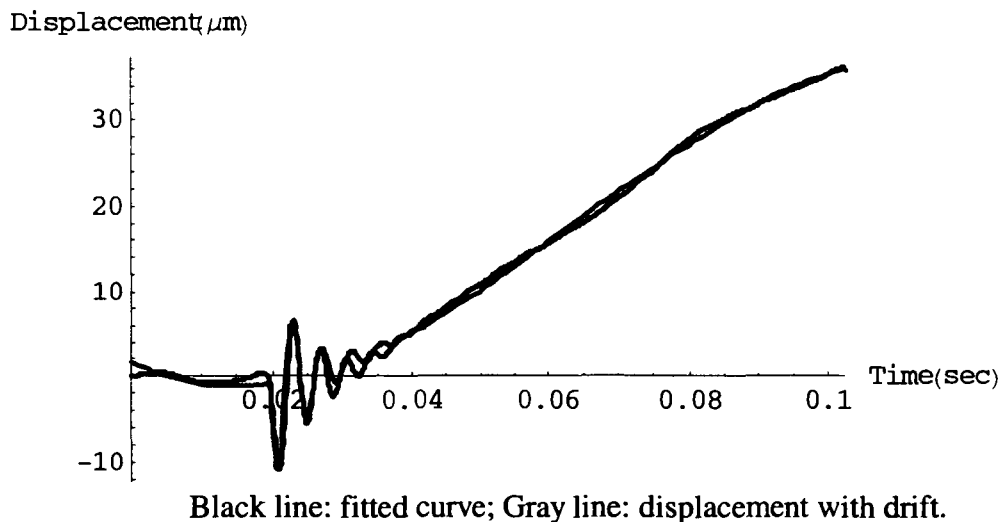
Figure 5.17 Curve fitting By using "MultiStartMin".

Black line: estimated impact-response; Gray line: displacement with drift.
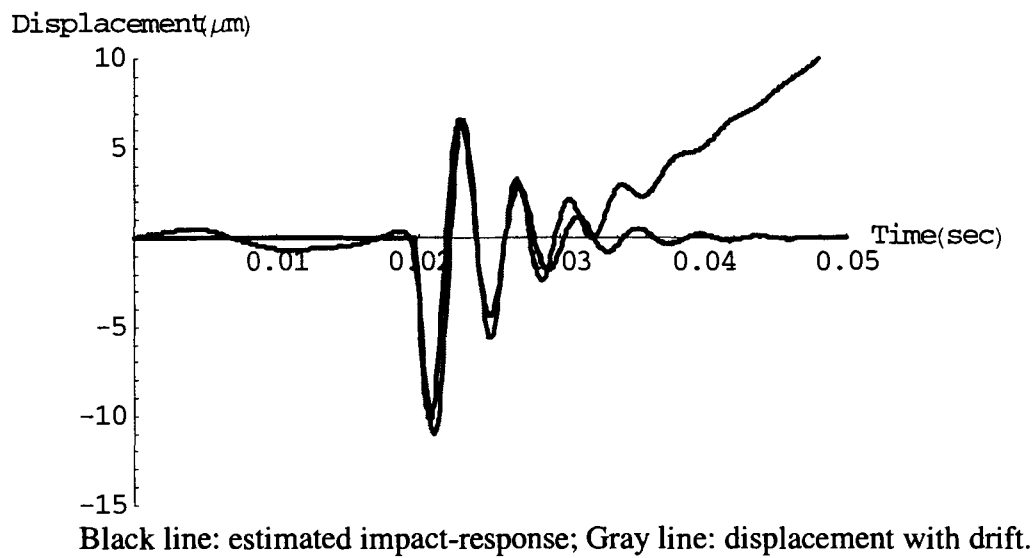
Figure 5.18 Comparison of the estimated impact-response and double integrated acceleration (displacement with drift).

6. APPLICATION TO VIBRATION VISUALIZATION

Entire process for visualization is partitioned into four major steps, namely, (1) data acquisition procedure, (2) signal processing, (3) generalized coordinates calculation, and (4) 3-D animation procedure. In these processing steps, (1) and (2) are already discussed in Chapter 5. So the step (3), and (4) are the subject to be discussed in this chapter.

6.1   VISUALIZATION OF SYSTEM VIBRATION

As assumed in the model of dynamometer, in this chapter, visualization is also derived under the assumption of "rigid body" of the plate of the dynamometer. There are six degrees of freedom (DOF) for the plate, namely three for translation, and three for rotation. With these variables, generalized coordinate list d_G, is formed as shown in Eq. 6.1, and Fig. 6.1 shows the coordinate components of the plate. With one generalized coordinate list, one specified position of the plate at a certain time instance can be described. And by showing these sequential pictures of generalized coordinate lists rapidly, the motion of system can be animated. The technique to calculate these generalized coordinate lists is discussed in this chapter.

$$\mathbf{d}_G = \begin{bmatrix} x & y & z & \theta & \phi & \psi \end{bmatrix}^T \tag{6.1}$$
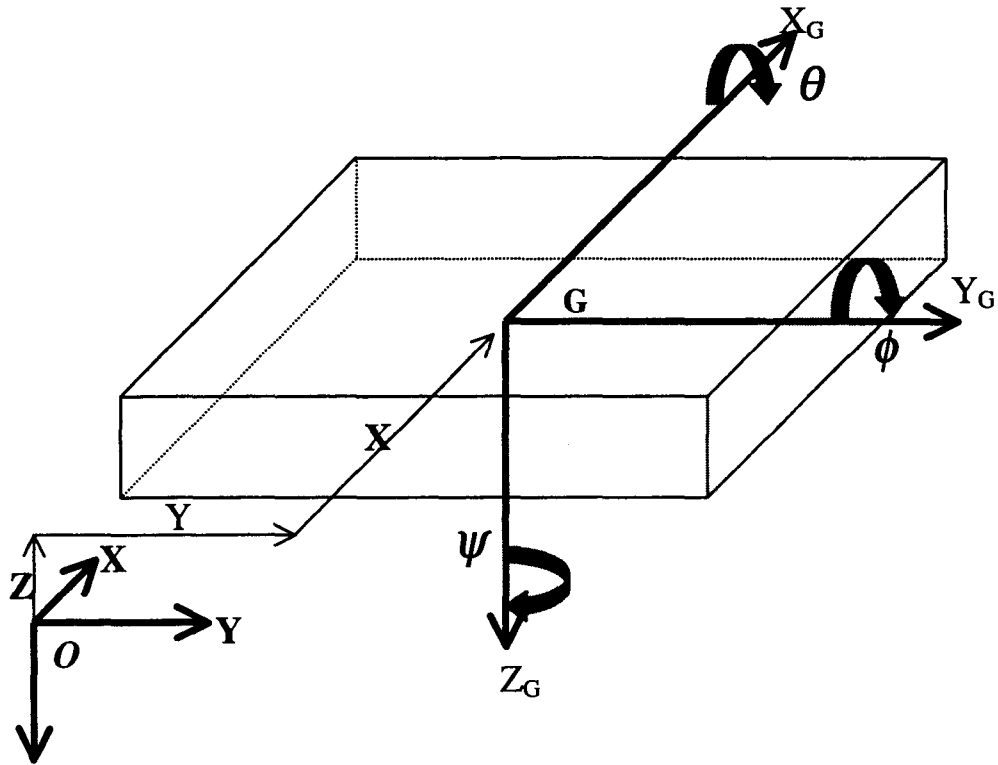
Fig. 6.1 Components of the generalized coordinate list $\mathbf{d_G}$, describing the 'rigid-body' motion of the plate.

The flowchart in Fig. 6.2 shows the algorithm used in this visualization. There are two different approaches. One is the signal based visualization. This means the visualization that is base on the experimental data. In this approach, the translational components are straightforward to calculate, but the rotational components should be calculated from the processed translational displacements signals by using the equations proposed by Padgaonkar et al. [1975], as further explained in section 6. The other is model based visualization. In this approach, six coordinate responses computed from the model derived in Chapter 4 are used for components of the generalized list $\mathbf{d_G}$. With these two generalized coordinate lists, animations of visualization are generated separately.

Impact

```
Physical
System
```

Accelerometers

Idealization and
Decomposition
Section 3.1

Data Acquisition
(LabVIEW® DAQ Controller)
Section 5.1

Simplified 'Rigid
Body' Model
Section 4.3

Signal Processing
-subtracting moving average
-conversion to physical units
-double integration procedure
-recovery of the "true" displcament
Section 5.3, 5.5

Simulating
Displacements
Section 4.5.3

Generalized Coordinate
Calculation
Section 6.3

3-D Animation
Section 6.4

Figure 6.2 Flowchart of the methodology used for the visualization of vibrations of
the dynamometer plate.

6.2   COORDINATE SYSTEM

Coordinate systems are defined in this chapter. A brief explanation of
terminology used henceforth is presented starting with the Coordinate Systems
(C.S.) as shown in Fig. 6.3.

1. Reference Coordinate System of the Plate $(XYZ)_R$

This is the coordinates system with initial position of the plate. Origin point of the C.S. is at the corner point marked C_R.

2. Instantaneous Coordinate System $(XYZ)_I$

This coordinates system moves together with vibrating plate. Points of corners remain without changing in this C.S., so this C.S. can be easily converted into the reference C.S., $(XYZ)_R$

3. Global Reference Coordinate System (XYZ)

The above coordinate systems suffice to visualize the motion of only one plate. However, if multiple plates are involved, it is advantageous to introduce one global coordinate system. With this better system, each plate has its unique Reference and Instantaneous coordinate systems.

The initial location of the reference C.S. can be described by a six components list, $\mathbf{d_o}$. First three components are the coordinates of C_R of Reference C.S. given in the Global Reference C.S. the three remaining components are angles between Reference C.S. and Global Reference C.S. The list, d_o is defined as

$$\mathbf{d_o} = \begin{bmatrix} x_o & y_o & z_o & \theta_o & \phi_o & \psi_o \end{bmatrix}^T \tag{6.2}$$
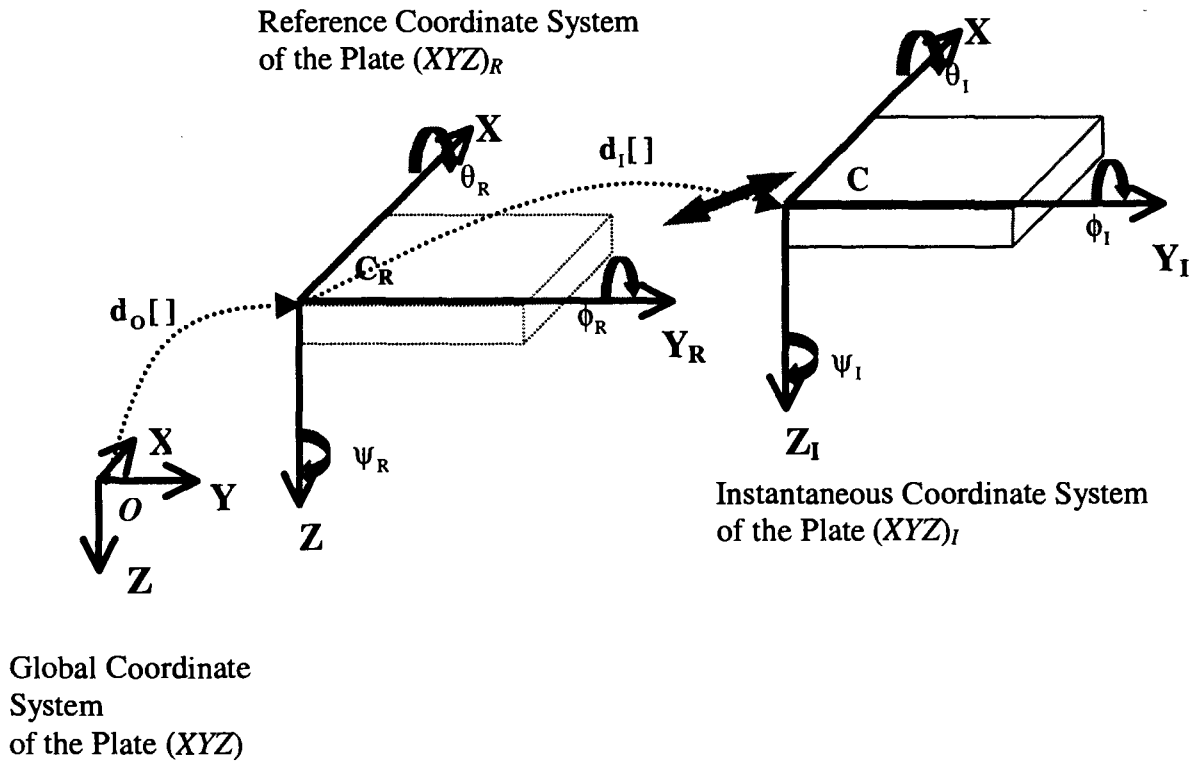
Figure 6.3 Coordinate systems used in describing the plate motion.

6.3   CALCULATION OF THE GENERALIZED COORDINATES

In physical systems, there are restrictions on where sensor can be mounted. For an example, locating the accelerometer at the system's center of mass may not be possible. However, the accelerometers can be easily mounted on flat surfaces of the plate, such as at point such as 'C' in Fig. 6.3. So it is advantageous to redefine the generalized coordinates describing the instantaneous spatial location of the plate. From now on consider the generalized coordinates that have their origin at the corner 'C' and then transform these to the coordinates that is defined at the center of mass by applying homogeneous coordinate transformation.

The first three components of the new list of generalized coordinates, d_I, define the position of point C in the reference C.S. $(XYZ)_R$. Three positions are

designated x_I, y_I, and z_I. The remaining three components define orientation of the plate with respect to $(XYZ)_R$. These later components are designated θ_1, ϕ_1, and ψ_1. Thus the entire new list of generalized coordinates is

$$\mathbf{d_I} = \begin{bmatrix} x_I & y_I & z_I & \theta_I & \phi_I & \psi_I \end{bmatrix}^T \tag{6.3}$$

In these generalized coordinates of instantaneous coordinate system, the rotations such as θ_1, ϕ_1, and ψ_1, are calculated according to a method proposed by Padgaonkar et al. (1975) briefly summarized below.

Under the rigid body assumption, the relative acceleration of a point P is given by the formula (Hibbeler, 1995).

$$a_P = a_C + \alpha \times r + \varpi \times (\varpi \times r) \tag{6.4}$$

where

p - the arbitrary point on a rigid plate, in this case one of the corners; P = 1, 2, and 3, shown in Fig. 6.4.

a_P - the acceleration of point P in $(XYZ)_R$,

a_C - the acceleration of point C in $(XYZ)_R$,

ϖ - the angular velocity of point P in $(XYZ)_I$

α - the angular acceleration of point P in $(XYZ)_I$, and

r - the position vector of point P from the origin, C, in $(XYZ)_I$.

From Eq.6.4, by substituting corresponding position numbers into P as 1, 2, and 3 with x, y, z direction, and C with x, y, and z direction, three nonlinear coupled equations are obtained as below [Padgaonkar et al., 1975].

$$\alpha_x = (a_{z1} - a_{zC})/r_y - \varpi_y \cdot \varpi_z \tag{6.4a}$$

$$\alpha_y = -(a_{z2} - a_{zC})/r_x + \varpi_x \cdot \varpi_z \qquad (6.4b)$$

$$\alpha_z = (a_{y2} - a_{yC})/r_x - \varpi_x \cdot \varpi_y \qquad (6.4c)$$

where

α_i - the angular acceleration component of the vector α around the i axis; $i =$ x,y,z,

ϖ_i - the angular velocity component of the vector ϖ around the i axis; $i = $ x,y,z.

r_x , r_y , r_z - the distances between accelerometers shown in Fig. 6.4.

As shown in equations 6.4a, 6.4b, and 6.4c, from the linear accelerations, the angular accelerations can be calculated. At minimum, six linear accelerations and three angular velocities are required. Six linear accelerations can be easily obtained from the accelerometers, which are shown as thick gray arrows in Fig 6.4, but the angular velocities are not so simple. With angular velocities, these nonlinear differential equations should be solved numerically. This is a time consuming task and the convergence of solution is not guaranteed. [Padgaonkar, 1975; Lie, 1976]. So there is another way to get the angular accelerations. With three more linear accelerations, which are captured from three accelerometers shown as black thick arrows in Fig 6.4, three more equations that are similar to equations 6.4a, 6.4b, and 6.4c, are obtained as

$$\alpha_x = (a_{y3} - a_{yC})/r_z + \varpi_y \cdot \varpi_z \qquad (6.4d)$$

$$\alpha_y = (a_{x3} - a_{xC})/r_z - \varpi_x \cdot \varpi_z \qquad (6.4e)$$

$$\alpha_x \qquad \qquad \qquad _3 \qquad (6.4f)$$
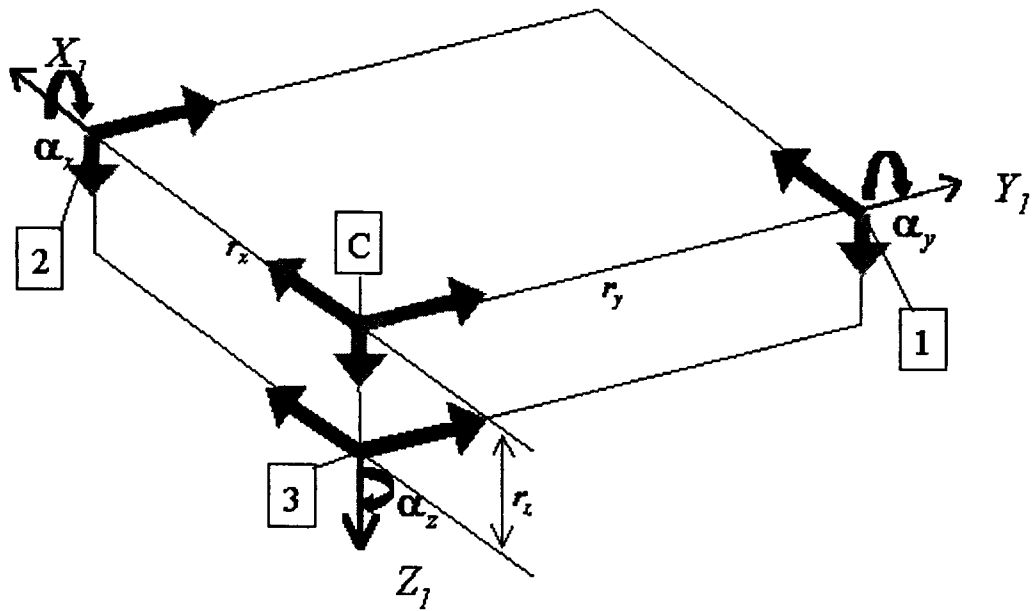
Figure 6.4 Locations of nine accelerometers required for the calculation of the generalized coordinates [Padgaonkar et al., 1975].

Now by eliminating cross products of the angular velocities from equations 6.4a –6.4f, the angular accelerations can be obtained as

$$\alpha_x = (a_{z1} - a_{zC})/2 \cdot r_y - (a_{y3} - a_{yC})/2 \cdot r_z \tag{6.5a}$$

$$\alpha_y = (a_{x3} - a_{xC})/2 \cdot r_z - (a_{z2} - a_{zC})/2 \cdot r_x \tag{6.5b}$$

$$\alpha_z = (a_{y2} - a_{yC})/2 \cdot r_x - (a_{x1} - a_{xC})/2 \cdot r_y \tag{6.5c}$$

Where

$$\alpha_x = \frac{d}{dt^2}[\theta_I(t)]^2 ; \alpha_y = \frac{d}{dt^2}[\phi_I(t)]^2 ; \alpha_z = \frac{d}{dt^2}[\psi_I(t)]^2 .$$

The roll (θ_I), pitch (ϕ_I), and yaw (ψ_I) angles are calculated using displacements obtained by double integration procedures from the all nine accelerometers on the plate according to equations.

$$\theta_I = (x_{z1} - x_{zC})/2 \cdot r_y - (x_{y3} - x_{yC})/2 \cdot r_z \qquad (6.6a)$$

$$\phi_I = (x_{x3} - x_{xC})/2 \cdot r_z - (x_{z2} - x_{zC})/2 \cdot r_x \qquad (6.6b)$$

$$\psi_I = (x_{y2} - x_{yC})/2 \cdot r_x - (x_{x1} - x_{xC})/2 \cdot r_y \qquad (6.6c)$$

where

$x_{i,j}$ - displacement obtained by double integration of the acceleration $a_{i,j}$,

i, j – the notation for axis, i = x, y, and z; j = C, 1, 2, and 3 (corners of the plate where the accelerometers are located).

The "true" displacements used above for the calculation of the angles are obtained by additional procedure such as "drift elimination" after the double integration as discussed in Chapter 5.

6.4   ANIMATION OF THE RIGID BODY MOTION

The animation procedure consists of three steps: (1) finding the absolute position of the reference corner points, C, (2) calculating coordinates of the plate's center in the C.S. $(XYZ)_R$ using homogeneous coordinate transformation, and drawing a single 3-D picture representing instantaneous position of the plate by using homogeneous transformation, and (3) animation of the 3-D pictures. These steps are discussed in the following subsections.

6.4.1   Finding Absolute Position of the Reference Corner Point

The coordinate vector $\mathbf{d}_I$ of the instantaneous coordinate system represents only the relative position and orientation of the C.S. $(XYZ)_I$ with respect to

$(XYZ)_R$ as shown in Fig.6.3. Absolute generalized coordinates of plate in position 'C' can be calculated by combining $\mathbf{d_O}$ (Eq. 6.2), and $\mathbf{d_I}$ (Eq. 6.3). List of the reference position, $\mathbf{d_O}$ can be assumed '0' because only one plate is being concerned in this research. And considering that the movements are small, a suitable magnification coefficients, such as K_t for translational movement, and K_r for rotational movement need to be introduced. So the absolute generalized coordinates $\mathbf{d_A}$ is calculated as

$$\mathbf{d_A} = \mathbf{d_O} + \mathbf{K \cdot d_I} \tag{6.7}$$

Where

$$x_A = K_t \cdot x_I, \qquad \theta_A = K_r \cdot \theta_I,$$

$$y_A = K_t \cdot y_I, \qquad \phi_A = K_r \cdot \phi_I,$$

$$z_A = K_t \cdot z_I, \qquad \psi_A = K_r \cdot \psi_I,$$

6.4.2   Homogeneous Transformation

For the visualization of vibrating plate, "solid" representation supported by *Mathematica* is used. Three programs are used for visualization in this research. The programs are written in *Mathematica*, mainly using homogeneous transformation matrices and as follow,

1. "visual.nb". This program develops the plate shape by connecting eight corners, shown as Eq. 6.8, and by homogeneous transformation matrices (Wovolich, 1987), calculates the location of the plate after the excitation. And by showing rapidly changing sequential plate shape, visualization of the vibrating plate is accomplished. The generalized coordinates at the center is required for this calculation.

2. "coordi_maker.nb". This program makes the generalized coordinate that is needed for the program "visual.nb". First the x, y, and z direction displacements, which are calculated from the signals of accelerometers, are read and the angles, θ_I , ϕ_I , and ψ_I are calculated as it is discussed in Section 6.3. Finally by homogeneous coordinate transformation technique, the generalized coordinates of center is obtained from the coordinate at the point 'C'.

3. "signal_process.nb". This program processes the raw signals from accelerometers to get the displacements. (Chapter 5)

Corners of plate of dynamometer are defined as

```
FRU[length_, width_, height_] = {0.5 width, -0.5 length, 0.5 height};
FLU[length_, width_, height_] = {-0.5 width, -0.5 length, 0.5 height};
FLL[length_, width_, height_] = {-0.5 width, -0.5 length, -0.5 height};
FRL[length_, width_, height_] = {0.5 width, -0.5 length, -0.5 height};
RRU[length_, width_, height_] = {0.5 width, 0.5 length, 0.5 height};
RLU[length_, width_, height_] = {-0.5 width, 0.5 length, 0.5 height};
RLL[length_, width_, height_] = {-0.5 width, 0.5 length, -0.5 height};
RRL[length_, width_, height_] = {0.5 width, 0.5 length, -0.5 height};
```
(6.8)

Homogeneous transformation matrices, which are used in this visualization are as follow.

1. 'Yaw' transformation matrix, $\mathbf{T_Y}(\phi_I)$.

This matrix accounts for a rotation around the Z_0 axis by an angle ψ .

$$\mathbf{T_Y}(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 & 0 \\ \sin(\psi) & \cos(\psi) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
(6.9)

2. 'Pitch' transformation matrix, $\mathbf{T_P}(\phi)$.

This matrix accounts for a rotation around the Y_1 axis by an angle ϕ.

$$T_P(\phi) = \begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(\phi) & 0 & \cos(\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (6.10)$$

3. 'Roll' transformation matrix, $T_R(\theta)$.

This matrix accounts for a rotation around the X_2 axis by an angle θ.

$$T_R(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & 0 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (6.11)$$

4. Translational transformation matrix, $T_T(x,y,z)$.

This matrix accounts for x, y, and z translations along the X_3, Y_3, and Z_3 axes.

$$T_T(x,y,z) = \begin{bmatrix} 0 & 0 & 0 & x \\ 0 & 0 & 0 & y \\ 0 & 0 & 0 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (6.12)$$

A succession of coordinate transformations, each represented by a particular transformation matrix, can be represented by a matrix product of individual transformation matrices [Wolovich 1987]. Therefore, the total transformation matrix that including the translational, and the rotational can be obtained as shown in Eq. 6.13.

$$T_{HT}(x,y,z,\theta,\phi,\psi) = T_T(x,y,z) \cdot T_R(\theta) \cdot T_P(\phi) \cdot T_Y(\psi) \qquad (6.13)$$

The position of point C in C.S. $(XYZ)_R$ is directly calculated by double integration procedure, and as shown in Fig. 6.5., the position can be described as vector, $V_O^C = [x_C, y_C, z_C]^T$.

Homogeneous coordinate transformation technique is applied to find coordinates of the point G, which is the system's center of mass. The coordinate vector of center G, $V_O^G = [x_G, y_G, z_G]^T$ is shown in Fig. 6.5. And the following steps are used to calculate the coordinates. First, vectors V_O^G and V_O^C are rearranged to the column form 1×4 as shown in Eq. 6.15, to be compatible with the homogeneous transformation matrix.
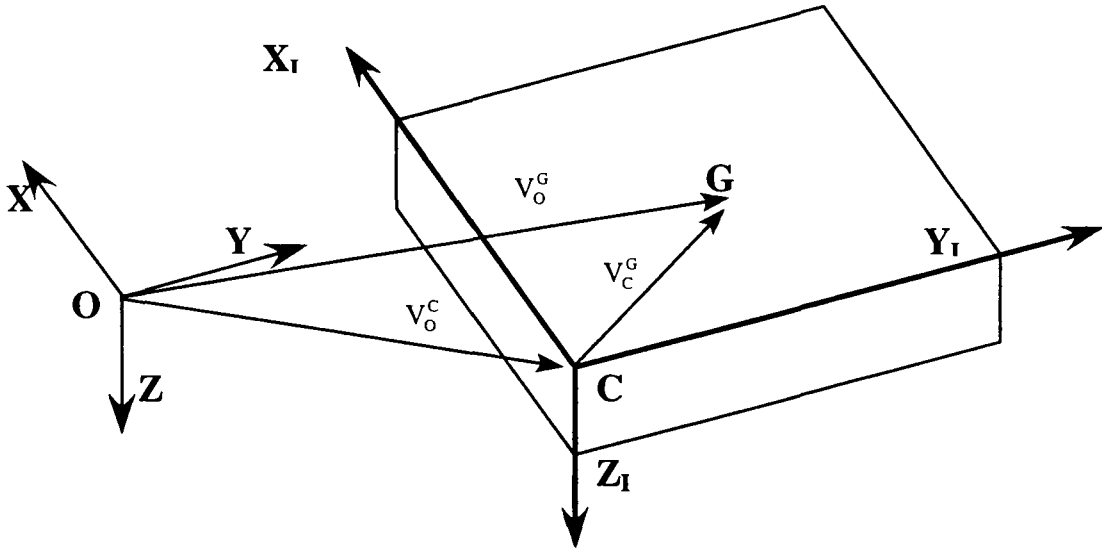
Figure 6.5 Application of the homogeneous coordinate transformation for finding coordinates of point G (system's center of mass).

$$D_O^C = \begin{bmatrix} V_O^C \\ 1 \end{bmatrix} = \begin{bmatrix} x_C \\ y_C \\ z_C \\ 1 \end{bmatrix} \quad (6.14), \quad \text{and} \quad D_O^G = \begin{bmatrix} V_O^G \\ 1 \end{bmatrix} = \begin{bmatrix} x_G \\ y_G \\ z_G \\ 1 \end{bmatrix} \quad (6.15)$$

It should be noted that the coordinates of point G defined in C.S. $(XYZ)_I$ are converted to its coordinates in the global C.S. (XYZ). A location vector, $\mathbf{V}_C^G$, is defined as a vector that contains local coordinates of the corner under consideration in the C.S. $(XYZ)_I$, *i.e.*, the coordinate with reference to the corner, C. In this example, dimensions of the plate are l, w, and h, where

l - the length of the plate measured parallel to the X_I axis,

w - the length of the plate measured parallel to the Y_I axis, and

h - the length of the plate measured parallel to the Z_I axis.

The column vector, $\mathbf{D}_C^G = [\ \dfrac{l}{2}\ \ \dfrac{w}{2}\ \ \dfrac{h}{2}\ \ 1\ ]^T$, arranged from the vector $\mathbf{V}_C^G$, is then used in the calculation of the coordinates of the point G in the global C.S. (XYZ) as

$$\mathbf{D}_O^G = \mathbf{T}_{HT}(x_C, y_C, z_C, \theta_C, \phi_C, \psi_C) \cdot \mathbf{D}_C^G \tag{6.16}$$

6.4.3   Drawing a 3-D Picture

To create a complete motion of the plate in a certain time, its edges (eight corner points) are plotted and connected with line by using 'Graphic3D' in *Mathematica*. The generalized coordinates including three translations and three rotations are required for the position of excited plate of dynamometer at a certain time. Consecutive plots of the plate are obtained by using the sequential generalized coordinates. And by showing these consecutive plots rapidly, the movement of the plate according to the actual vibration or according to the simulation of model is accomplished as shown in Fig. 6.6.
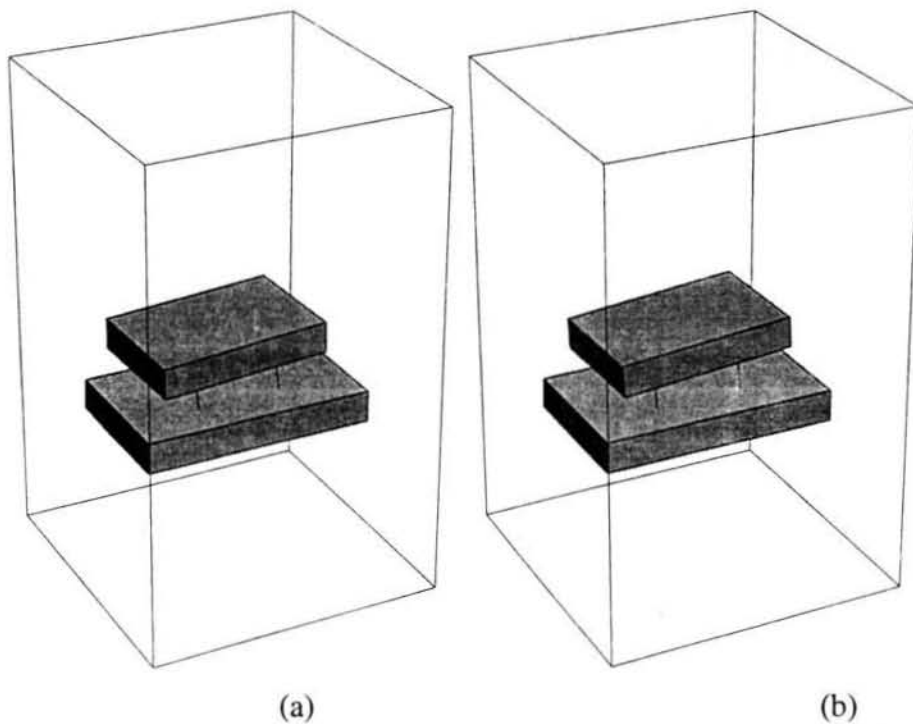
Figure 6.6 Captured pictures of the animated dynamometers. (a) from actual signal, (b) from the model.

6.5   CLOSURE

Two kinds of 3-D visualizations for the vibrating dynamometer are performed. After developing the visualization program, 'visual.nb' by using *mathematica*, calculating the generalized coordinates turns to be main concerns. From the simulated model, the generalized coordinates are easily obtained because the coordinates are same between the model and the animation. Different from the model, several procedures are required to calculate the generalized coordinates from the actual signal from the accelerometers. The rotations should be calculated from the translations, and the transformation is also needed to get the generalized coordinates of the center. By these different procedures, two visualizations are accomplished and compared as shown in Fig. 6.6. Visual comparison of animated

motion in the case of different excitation signals showed a good agreement of the model based and experimentally obtained responses of the platform. It corroborated the conclusion from Chapter 5 that the developed model of the dynamometer well represents its behaviors.

7. CONCLUSIONS AND RECOMMENDATIONS

7.1 CONCLUSIONS

The presented Computer-Aided Model Generation of a High performance Dynamometer gives the methodology of analyzing the dynamic system with multi-degree of freedom. Generation of the system's model is the first stage. And the validation of the model is the second stage. In the first stage, the rigid body assumption is made for the modeling in consideration that the deformation of dynamometer's plate is negligible. For the simplification purpose, the fixed foundation and base assumption is also applied. Finally six transfer functions that represent the motion of the platform are obtained by using Lagrange's energy formalism. The use of *"Mathematica"* with the *"Control System Professinal"* package provides several ways to check the result of the modeling. By plotting 'bode plot's of the each transfer functions, the natural frequencies of the system can be obtained. Also the output responses of the system can be simulated. The second stage consists of data processing and visualization procedures. Signals captured from the accelerometers are double integrated to get the displacements. Several extra processes are applied to eliminate drifts. For the strong signals, drift elimination procedure is simple and easy, but for the weak signals, it is not as easy to eliminate drift as it is for the strong signals. Several minimization techniques are discussed and acceptable drift estimation is achieved. The comparison between the displacements calculated from the signals and the displacements simulated from the model provides reasonable matches. For the application of this research, the visualization is performed base on the both ways, namely signal based one, and model based one. The generalized coordinates, which are necessary for the animation of vibrating platform is calculated. As expected from the comparison of the displacements, the animations of both ways matched quit well.

7.2 RECOMMENDATIONS FOR FURTHER RESEARCH

To accomplish the complete model with more accuracy, and more efficiency way, additional works are required. For the model generation, reconsideration of the assumptions would improve the results. For an example, employing flexible mode or combination of rigid body and flexible mode would be taken into account to get the closer result from the actual system. And expanding the degree-of-freedom by considering the movements of base, instead of fixed base assumption model would provide the closer model to the actual system.

Validation for the improved model also should be improved. For the signal processing procedures, elimination of drifts should be performed with the advanced methods because the errors considered as negligible in this research would not be negligible in further research. For the model with flexible mode, visualization should be considered differently from the model with rigid body. Employing platform with several sections would be appropriate for the model with flexible mode. To accomplish this, more sensors can be employed and a modal analysis technique would be applied.

BIBLIOGRAPHY

Aini, R., Rahnejat, H., and Gohar, R., "A Five Degrees of Freedom Analysis of Vibrations in Precision Spindles.", Int. Journal of Machine Tool and Manufacturing, Vol. 30, No. 1, pp. 1-18, 1990.

Baratange, François, "Enhanced Sensor for Vehicle Dynamics Evaluation", M.S. Thesis, Oregon State University, 1999.

Bhatti, M. Asghar, Practical optimization methods : with Mathematica application, Th Electronic Library of Science, 1998.

Brisbone, Brian P. "Computer-Aided Model Generation and Validation For Dynamic Systems", M.S. Thesis, Oregon State University, 1998.

Carne, T. G. et. Al., "Finite Element Analysis and Modal Testing of a Rotating Wind Turbine", International Journal of Analytic and Experimental Modal Analysis, Vol. 3, p.33,1988.

Cheung, Y. K., Leung, A. Y. T., Finite Eliment Methods in Dynamics, Science Press, Kluwer Academic Publishers, 1991.

Cheung, Y., "Model Based Adaptive Compensation of Dynamic Characteristic of In-Process Sensors", Ph.D. Thesis, University of Wisconsin-Madison, 1993.

Comparin, R. J., "The Dynamic Analysis of a Machine Tool Spindle", M.S. Thesis, Ohio State University, 1983.

Doebelin, Ernest O., Measurement System : application and design, 4th ed., McGRAW-HILL, 1990.

Ewins, D.J., Modal Test: Theory and Practice, John Wiley & Sons, Inc., 1984.

Fagan, M.J., Finite Element Analysis: Theory and Practice, Longman Scientific & Technical, Wiley Harlow, 1992.

Franklin, Gene F., Powell, J. David, Emami-Naeini, A., Feedback Control of Dynamic Systems, 3th ed., Addison-Wesley Publishing Company, Inc. 1994

Friswell, M.I., Mottershead, J.E., Finite Model Updating in Structural Mechanics, Kluwer Academic Publishers, 1995.

Ginsberg, J.H., <u>Advanced Engineering Dynamics</u>, Cambridge University Press, 1995.

Hale, A. L., Meirovitch, L., "Derivation of the Equations of Motion for Complex Structures by Symbolic Manipulation", Computers and Structures, Vol. 9, pp. 639-649, 1978.

Hansen, Eldon, <u>Global Optimization Using Interval Analysis</u>, Marcel Dekker, Inc., 1992.

Hibbeler, R. C. <u>Engineering mechanics-Statics and Dynamics</u>. 7th ed. New Jersey: Prentice-Hall, Inc., 1995.

Jitpraphai, T., "Model Based Visualization of Vibrations in Mechanical Systems", M.S. Thesis, Oregon State University, 1997.

Kane, T. R., Levinson, D. A., <u>Dynamics: Theory and Applications</u>, McGraw-Hill, Inc., 1985.

Karnopp, D., Rosenberg, R., <u>System Dynamics: A Unified Approach</u>, John Wiley & Sons, Inc., 1975.

Lieh, J., Haque, I., "Symbolic Closed-Form Modeling and Linearization of Multibody systems subject to Control", Journal of Mechanical Design, Vol. 113, pp. 124-132, 1991.

Matsubara, M., et. al., "Computational Modeling of Precision Spindles Supported by Ball Bearing", Int. Journal of Machine Tool and Manufacturing, Vol. 28, No. 4, pp. 429-442, 1988.

Nise, Norman S., <u>Control Systems Engineering</u>, 2nd ed., The Benjamin/Cummings Publishing Company, Inc. 1995

Ogata, Katsuhiko, <u>Discrete-Time Control Systems</u>, Prentice-Hall, 1987.

Ogata, Katsuhiko, <u>System Dynamics</u>, 2nd ed., Prentice-Hall, 1992.

Padgaonkar, A. J., Krieger, K. W. and Kings, A. I., "Measurement of Angular Acceleration of a Rigid Body Using Linear Accelerometers." <u>Journal of Applied Mechanical Transactions of the ASME</u>, Vol. 42, pp. 552-556, 1975.

Pandit, S. M., <u>Modal and Spectrum Analysis: Data Dependent Systems in State Space</u>, John Wiley & Sons, Inc., 1991

Reddy, V. R., Sharan, A. M., "The Finite Element Modeled Design of Lathe Spindles: The Static and Dynamic Analysis", ASME Journal of Vibrations, Acoustics, Stress and Reliability in Design, Vol. 109, pp. 407-415, 1987.

Rosenthal, D. E., Sherman, M. A., "High Performance Multibody Simulations via Symbolic Equation Manipulation and Kane's Method", Journal of the Astronautical Sciences, Vol. 34, No. 3, pp. 223-239, 1986.

Scheck, F., Mechanics, Springer-Verlag, 1994.

Shabana, A. A., Theory of Vibration, Springer-Verlag, 1991.

Shin, Y. C., Wang, K. W., Chen, C. H., "Dynamic Analysis and Modeling of a High Speed Spindle System", Transactions of NAMRI.SME, pp. 298-304, 1990.

Spiewak, S. A., "Modeling of Cutting Point Trajectories in Milling.", ASME

Spiewak, S. A., "Acceleration Based Indirect Force Measurement in Metal Cutting Processes", International Journal of Machine Tools and Manufacturing, Vol. 35, No. 1, pp. 1-17, 1995.

Weaver, W. Jr., Johnston, P. R., Structural Dynamics by Finite Elements, Prentice-Hall, 1987.

Weck, M., Handbook of Machine Tools, John Wiley, 1984.

Weikert, S., Rehsteiner, F., Brisbine, B., Spiewak, S. A., "Model Based Minimization of Dynamic Tool Path Errors", Proc. Second SAE Aerospace Manufacturing Technology Conference, Long Beach CA, 1998.

APPENDICES

Appendix A
Experimental Specifications, and Constant Parameters.

A.1 Information of Sensor used in the Experiment

Comments	Direction in Sensor	Location (Fig.)	Sensitivity	Channel	Amplifier Gain
ADXL202	x_h	a_{cx}	0.312 V/g	2	1
ADXL202	y_h	a_{cy}	0.312 V/g	3	1
ADXL202	y_v	a_{cz}	0.312 V/g	4	1
ADXL202	y_1	a_{1y}	0.312 V/g	5	1
ADXL202	x_1	a_{1z}	0.312 V/g	6	1
ADXL202	x_2	a_{2x}	0.312 V/g	7	1
ADXL202	y_2	a_{2z}	0.312 V/g	8	1
ADXL202	y_3	a_{3x}	0.312 V/g	9	1
ADXL202	x_3	a_{3y}	0.312 V/g	10	1
Kistler 8302A10/5202A2		a_{1y}	0.5 V/g	11	1
Load cell		Not fixed	2.248 mV/N	12	10

A.2 Dimensions of the Dynamometer
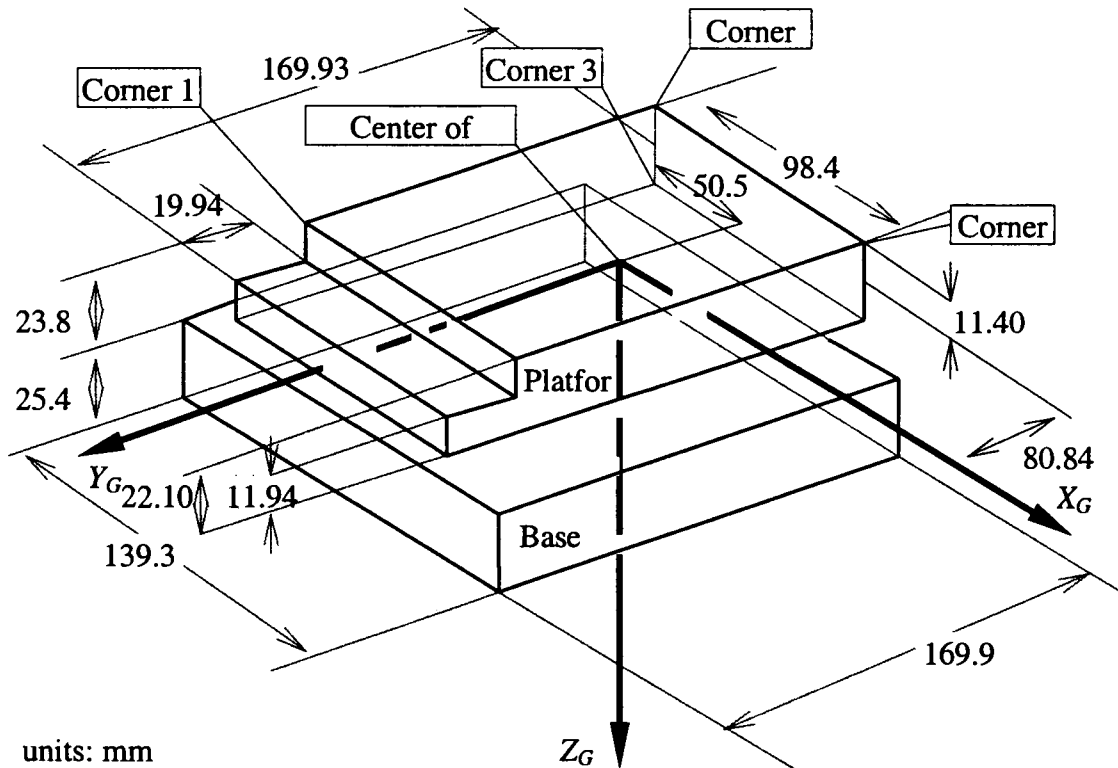
Figure A.1 Dimensions of the dynamometer used in the experiment in units of mm
(the sensing elements are not shown).[Jitpraphai, 1997]

A.3 Constant Parameters in the Spatial Matrices

Mass of the platform [Chen, 1996].

m = 2.714 kg.

Mass moment of inertia of the platform around the center of mass, G [Chen, 1996].

Ixx = .006187217392 kg·m^2, Iyy = .002398407627 kg·m^2, Izz = .008366544210

kg·m^2,

Ixy = 0 kg·m^2, Iyz = .00007500541682 kg·m^2, Ixz = 0 kg·m^2.

Stiffness of the sensing elements [Chen, 1996].

$k01 = 262000$ N/m, $k02 = 1260000$ N/m.

$k01 = kx1 = kx2 = kx3 = kx4 = ky1 = ky2 = ky3 = ky4$.

$k02 = kz1 = kz2 = kz3 = kz4$.

Damping coefficients of the sensing elements [Chen, 1996].

$cox = 140$ N·s/m, $coy = 125$ N·s/m, $coz = 275$ N·s/m.

$cox = cx1 = cx2 = cx3 = cx4$.

$coy = cy1 = cy2 = cy3 = cy4$.

$coz = cz1 = cz2 = cz3 = cz4$.

Distances of the sensing elements from the center of mass [Chen, 1996].

$a = 0.02999$ m (in X_G direction),

$b = 0.04979$ m (in Y_G direction),

$h = 0.01070$ m (in Z_G direction).

Appendix B
Plots of the Experimental Responses from 'y' direction impact force.
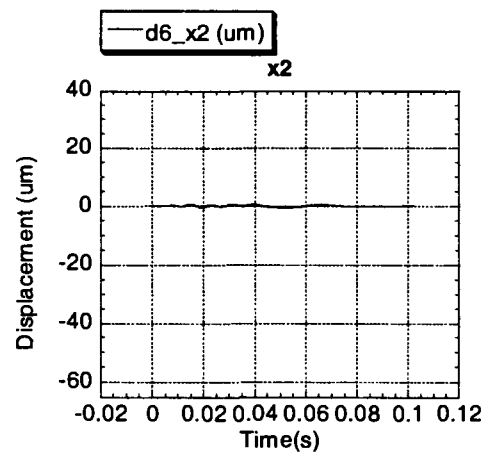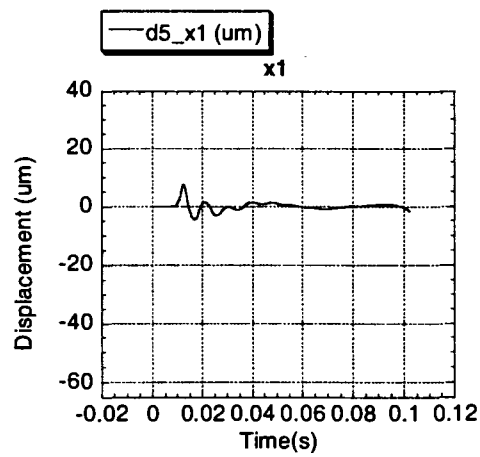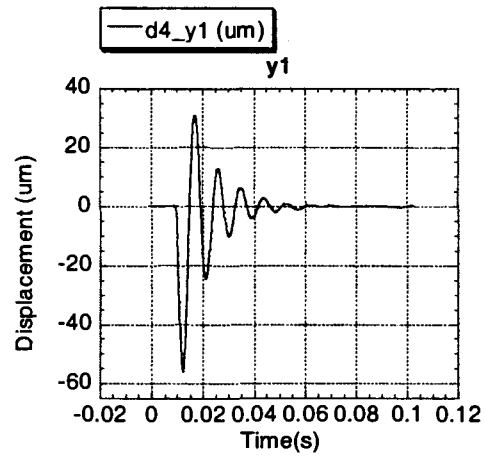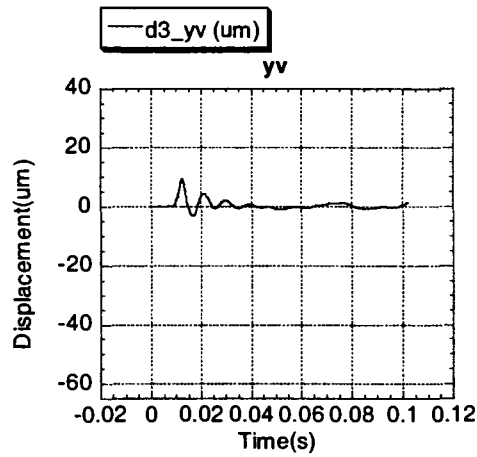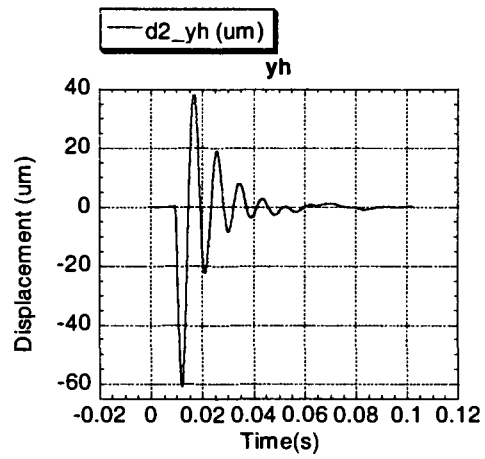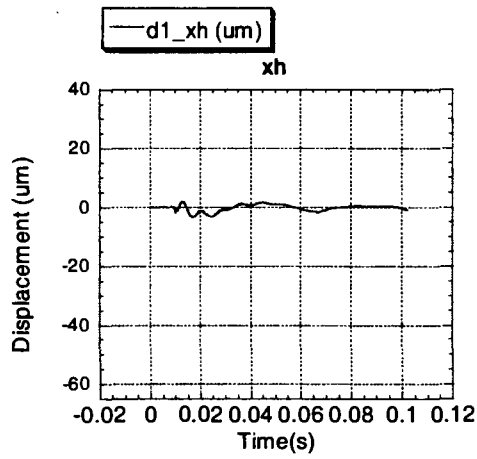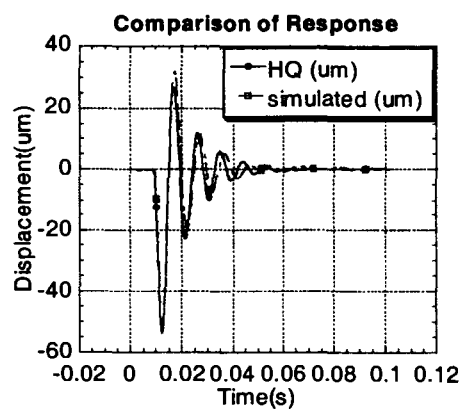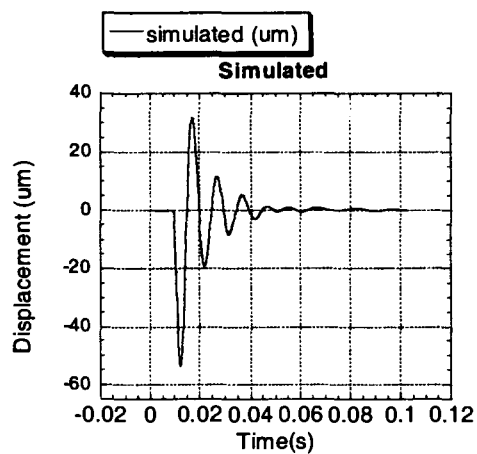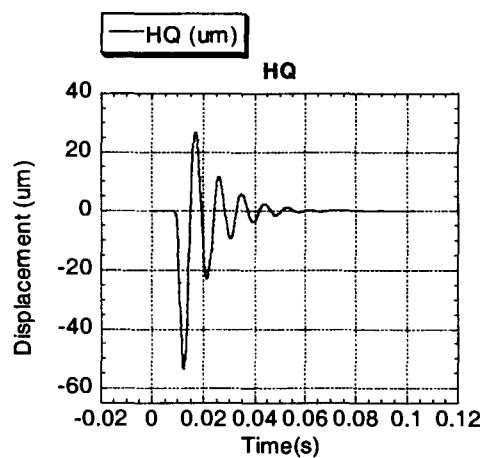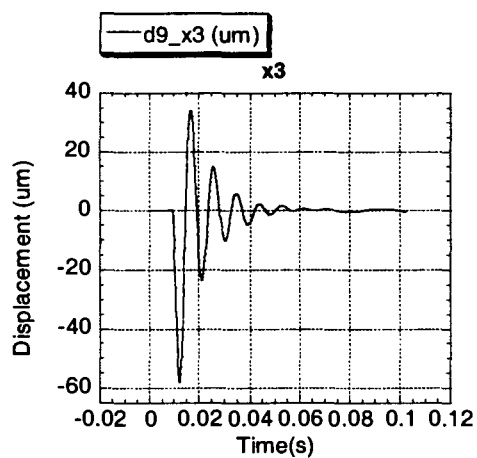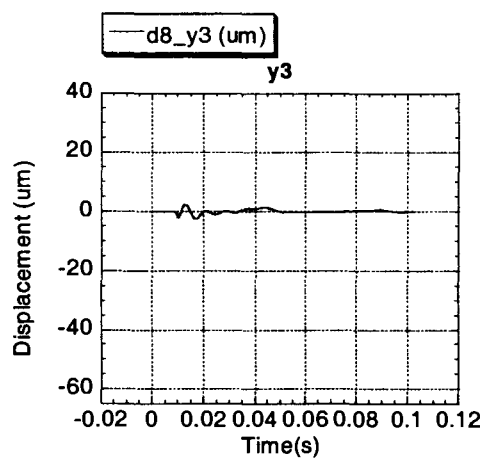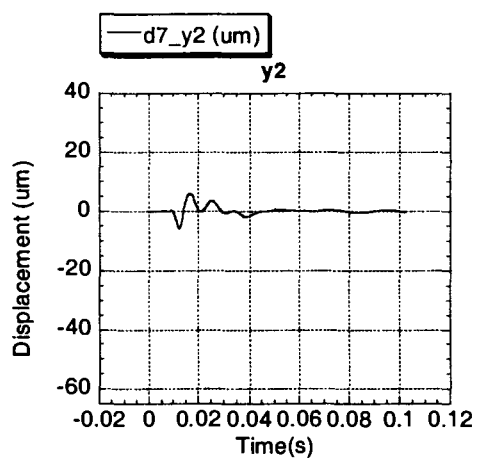
Appendix C
Model Derivation of Dynamometer in *Mathematica*

Model of Dynamometer
Generating Equations of Motion

Development History:

Based on a model of HEXACT ("Hexact5_3.nb") developed by B. Brisbine (last rev. 05/03/99)

Started:	**7/11/99:**	**S. Ko**
Last Rev.:	**12/5/2000:**	**S. Ko**

0. Call packages (Check if loaded succesfully)
& clear numerical constants and variables
1. Set properties of the working environment
2. Symbolize variables
3. Vectors of coordinates and system parameters
4. Transformation matrices
5. Defining EDE position vectors
6. Potential Energy
7. Kinetic energy
8. Damping energy
9. External input force and moment
10. The Lagrange's equations of motion
10.1 Define the Lagrangian function.

```
LagrEqns[T_, U_, Damp_, Q_List, Coor_List] :=
    Flatten[MapThread[{∂t (∂∂t#2 T) - ∂#2 (T - U) + ∂∂t#2 Damp - #1 == 0} &, {Q, Coor}]];
```

10.2 Calculate equations of motion.
incrCoord
{x1(*t*), y1(*t*), z1(*t*), θ1(*t*), φ1(*t*), ψ1(*t*), x2(*t*), y2(*t*), z2(*t*), θ2(*t*), φ2(*t*), ψ2(*t*)}

EOM1 = LagrEqns [T$_t$, U$_t$, D$_t$, minGen , incrCoord];

Twelve equations of motion for the Dynamometer.

$$\Big\{ k_{pX3} \Big(x1(t) - x2(t) + \frac{11\,\phi 1(t)}{1000} - \frac{347\,\phi 2(t)}{20000} + \frac{\psi 1(t)}{20} - \frac{4979\,\psi 2(t)}{100000} \Big) +$$

$$k_{pX4} \Big(x1(t) - x2(t) + \frac{11\,\phi 1(t)}{1000} - \frac{347\,\phi 2(t)}{20000} + \frac{\psi 1(t)}{20} - \frac{4979\,\psi 2(t)}{100000} \Big) +$$

$$k_{pX1} \Big(x1(t) - x2(t) + \frac{11\,\phi 1(t)}{1000} - \frac{347\,\phi 2(t)}{20000} - \frac{\psi 1(t)}{20} + \frac{4979\,\psi 2(t)}{100000} \Big) +$$

$$k_{pX2} \Big(x1(t) - x2(t) + \frac{11\,\phi 1(t)}{1000} - \frac{347\,\phi 2(t)}{20000} - \frac{\psi 1(t)}{20} + \frac{4979\,\psi 2(t)}{100000} \Big) - f_x(t) +$$

$$b_{pX3}\left(x1'(t)-x2'(t)+\frac{11\,\phi1'(t)}{1000}-\frac{347\,\phi2'(t)}{20000}+\frac{\psi1'(t)}{20}-\frac{4979\,\psi2'(t)}{100000}\right)+$$

$$b_{pX4}\left(x1'(t)-x2'(t)+\frac{11\,\phi1'(t)}{1000}-\frac{347\,\phi2'(t)}{20000}+\frac{\psi1'(t)}{20}-\frac{4979\,\psi2'(t)}{100000}\right)+$$

$$b_{pX1}\left(x1'(t)-x2'(t)+\frac{11\,\phi1'(t)}{1000}-\frac{347\,\phi2'(t)}{20000}-\frac{\psi1'(t)}{20}+\frac{4979\,\psi2'(t)}{100000}\right)+$$

$$b_{pX2}\left(x1'(t)-x2'(t)+\frac{11\,\phi1'(t)}{1000}-\frac{347\,\phi2'(t)}{20000}-\frac{\psi1'(t)}{20}+\frac{4979\,\psi2'(t)}{100000}\right)+mP\,x1''(t)==0,$$

$$k_{pY2}\left(y1(t)-y2(t)-\frac{11\,\theta1(t)}{1000}+\frac{347\,\theta2(t)}{20000}+\frac{3\,\psi1(t)}{100}-\frac{2999\,\psi2(t)}{100000}\right)+$$

$$k_{pY3}\left(y1(t)-y2(t)-\frac{11\,\theta1(t)}{1000}+\frac{347\,\theta2(t)}{20000}+\frac{3\,\psi1(t)}{100}-\frac{2999\,\psi2(t)}{100000}\right)+$$

$$k_{pY1}\left(y1(t)-y2(t)-\frac{11\,\theta1(t)}{1000}+\frac{347\,\theta2(t)}{20000}-\frac{3\,\psi1(t)}{100}+\frac{2999\,\psi2(t)}{100000}\right)+$$

$$k_{pY4}\left(y1(t)-y2(t)-\frac{11\,\theta1(t)}{1000}+\frac{347\,\theta2(t)}{20000}-\frac{3\,\psi1(t)}{100}+\frac{2999\,\psi2(t)}{100000}\right)-f_y(t)+$$

$$b_{pY2}\left(y1'(t)-y2'(t)-\frac{11\,\theta1'(t)}{1000}+\frac{347\,\theta2'(t)}{20000}+\frac{3\,\psi1'(t)}{100}-\frac{2999\,\psi2'(t)}{100000}\right)+$$

$$b_{pY3}\left(y1'(t)-y2'(t)-\frac{11\,\theta1'(t)}{1000}+\frac{347\,\theta2'(t)}{20000}+\frac{3\,\psi1'(t)}{100}-\frac{2999\,\psi2'(t)}{100000}\right)+$$

$$b_{pY1}\left(y1'(t)-y2'(t)-\frac{11\,\theta1'(t)}{1000}+\frac{347\,\theta2'(t)}{20000}-\frac{3\,\psi1'(t)}{100}+\frac{2999\,\psi2'(t)}{100000}\right)+$$

$$b_{pY4}\left(y1'(t)-y2'(t)-\frac{11\,\theta1'(t)}{1000}+\frac{347\,\theta2'(t)}{20000}-\frac{3\,\psi1'(t)}{100}+\frac{2999\,\psi2'(t)}{100000}\right)+mP\,y1''(t)==0,$$

$$z1''(t)\,mP+\frac{981\,mP}{100}+k_{pZ1}\left(z1(t)-z2(t)+\frac{\theta1(t)}{20}-\frac{4979\,\theta2(t)}{100000}+\frac{3\,\phi1(t)}{100}-\frac{2999\,\phi2(t)}{100000}\right)+$$

$$k_{pZ4}\left(z1(t)-z2(t)-\frac{\theta1(t)}{20}+\frac{4979\,\theta2(t)}{100000}+\frac{3\,\phi1(t)}{100}-\frac{2999\,\phi2(t)}{100000}\right)+$$

$$k_{pZ2}\left(z1(t)-z2(t)+\frac{\theta1(t)}{20}-\frac{4979\,\theta2(t)}{100000}-\frac{3\,\phi1(t)}{100}+\frac{2999\,\phi2(t)}{100000}\right)+$$

$$k_{pZ3}\left(z1(t)-z2(t)-\frac{\theta1(t)}{20}+\frac{4979\,\theta2(t)}{100000}-\frac{3\,\phi1(t)}{100}+\frac{2999\,\phi2(t)}{100000}\right)-f_z(t)+$$

$$b_{pZ1}\left(z1'(t)-z2'(t)+\frac{\theta1'(t)}{20}-\frac{4979\,\theta2'(t)}{100000}+\frac{3\,\phi1'(t)}{100}-\frac{2999\,\phi2'(t)}{100000}\right)+$$

$$b_{pZ4}\left(z1'(t)-z2'(t)-\frac{\theta1'(t)}{20}+\frac{4979\,\theta2'(t)}{100000}+\frac{3\,\phi1'(t)}{100}-\frac{2999\,\phi2'(t)}{100000}\right)+$$

$$b_{pZ2}\left(z1'(t)-z2'(t)+\frac{\theta1'(t)}{20}-\frac{4979\,\theta2'(t)}{100000}-\frac{3\,\phi1'(t)}{100}+\frac{2999\,\phi2'(t)}{100000}\right)+$$

$$b_{pZ3}\left(z1'(t)-z2'(t)-\frac{\theta1'(t)}{20}+\frac{4979\,\theta2'(t)}{100000}-\frac{3\,\phi1'(t)}{100}+\frac{2999\,\phi2'(t)}{100000}\right)==0,$$

$$\frac{1}{20}k_{pZ1}\left(z1(t)-z2(t)+\frac{\theta1(t)}{20}-\frac{4979\,\theta2(t)}{100000}+\frac{3\,\phi1(t)}{100}-\frac{2999\,\phi2(t)}{100000}\right)+$$

$$\frac{1}{20}k_{pZ3}\left(-z1(t)+z2(t)+\frac{\theta1(t)}{20}-\frac{4979\,\theta2(t)}{100000}+\frac{3\,\phi1(t)}{100}-\frac{2999\,\phi2(t)}{100000}\right)+$$

$$\frac{1}{20}k_{pZ2}\left(z1(t)-z2(t)+\frac{\theta1(t)}{20}-\frac{4979\,\theta2(t)}{100000}-\frac{3\,\phi1(t)}{100}+\frac{2999\,\phi2(t)}{100000}\right)+$$

$$\frac{1}{20}k_{pZ4}\left(-z1(t)+z2(t)+\frac{\theta1(t)}{20}-\frac{4979\,\theta2(t)}{100000}-\frac{3\,\phi1(t)}{100}+\frac{2999\,\phi2(t)}{100000}\right)-$$

$$\frac{1}{1000}\left(11\,k_{\mathrm{pY2}}\!\left(\mathrm{y1}(t)-\mathrm{y2}(t)-\frac{11\,\theta 1(t)}{1000}+\frac{347\,\theta 2(t)}{20000}+\frac{3\,\psi 1(t)}{100}-\frac{2999\,\psi 2(t)}{100000}\right)\right)-$$

$$\frac{1}{1000}\left(11\,k_{\mathrm{pY3}}\!\left(\mathrm{y1}(t)-\mathrm{y2}(t)-\frac{11\,\theta 1(t)}{1000}+\frac{347\,\theta 2(t)}{20000}+\frac{3\,\psi 1(t)}{100}-\frac{2999\,\psi 2(t)}{100000}\right)\right)-$$

$$\frac{1}{1000}\left(11\,k_{\mathrm{pY1}}\!\left(\mathrm{y1}(t)-\mathrm{y2}(t)-\frac{11\,\theta 1(t)}{1000}+\frac{347\,\theta 2(t)}{20000}-\frac{3\,\psi 1(t)}{100}+\frac{2999\,\psi 2(t)}{100000}\right)\right)-$$

$$\frac{1}{1000}\left(11\,k_{\mathrm{pY4}}\!\left(\mathrm{y1}(t)-\mathrm{y2}(t)-\frac{11\,\theta 1(t)}{1000}+\frac{347\,\theta 2(t)}{20000}-\frac{3\,\psi 1(t)}{100}+\frac{2999\,\psi 2(t)}{100000}\right)\right)+z_{\mathrm{in}}\,f_y(t)-$$

$$y_{\mathrm{in}}\,f_z(t)-m_x(t)+\frac{1}{20}\,b_{\mathrm{pZ1}}\!\left(\mathrm{z1}'(t)-\mathrm{z2}'(t)+\frac{\theta 1'(t)}{20}-\frac{4979\,\theta 2'(t)}{100000}+\frac{3\,\phi 1'(t)}{100}-\frac{2999\,\phi 2'(t)}{100000}\right)-$$

$$\frac{1}{20}\,b_{\mathrm{pZ4}}\!\left(\mathrm{z1}'(t)-\mathrm{z2}'(t)-\frac{\theta 1'(t)}{20}+\frac{4979\,\theta 2'(t)}{100000}+\frac{3\,\phi 1'(t)}{100}-\frac{2999\,\phi 2'(t)}{100000}\right)+$$

$$\frac{1}{20}\,b_{\mathrm{pZ2}}\!\left(\mathrm{z1}'(t)-\mathrm{z2}'(t)+\frac{\theta 1'(t)}{20}-\frac{4979\,\theta 2'(t)}{100000}-\frac{3\,\phi 1'(t)}{100}+\frac{2999\,\phi 2'(t)}{100000}\right)-$$

$$\frac{1}{20}\,b_{\mathrm{pZ3}}\!\left(\mathrm{z1}'(t)-\mathrm{z2}'(t)-\frac{\theta 1'(t)}{20}+\frac{4979\,\theta 2'(t)}{100000}-\frac{3\,\phi 1'(t)}{100}+\frac{2999\,\phi 2'(t)}{100000}\right)-$$

$$\frac{1}{1000}\left(11\,b_{\mathrm{pY2}}\!\left(\mathrm{y1}'(t)-\mathrm{y2}'(t)-\frac{11\,\theta 1'(t)}{1000}+\frac{347\,\theta 2'(t)}{20000}+\frac{3\,\psi 1'(t)}{100}-\frac{2999\,\psi 2'(t)}{100000}\right)\right)-$$

$$\frac{1}{1000}\left(11\,b_{\mathrm{pY3}}\!\left(\mathrm{y1}'(t)-\mathrm{y2}'(t)-\frac{11\,\theta 1'(t)}{1000}+\frac{347\,\theta 2'(t)}{20000}+\frac{3\,\psi 1'(t)}{100}-\frac{2999\,\psi 2'(t)}{100000}\right)\right)-$$

$$\frac{1}{1000}\left(11\,b_{\mathrm{pY1}}\!\left(\mathrm{y1}'(t)-\mathrm{y2}'(t)-\frac{11\,\theta 1'(t)}{1000}+\frac{347\,\theta 2'(t)}{20000}-\frac{3\,\psi 1'(t)}{100}+\frac{2999\,\psi 2'(t)}{100000}\right)\right)-$$

$$\frac{1}{1000}\left(11\,b_{\mathrm{pY4}}\!\left(\mathrm{y1}'(t)-\mathrm{y2}'(t)-\frac{11\,\theta 1'(t)}{1000}+\frac{347\,\theta 2'(t)}{20000}-\frac{3\,\psi 1'(t)}{100}+\frac{2999\,\psi 2'(t)}{100000}\right)\right)+\mathrm{JP}_{\mathrm{xx}}\,\theta 1''(t)==0,$$

$$\frac{3}{100}\,k_{\mathrm{pZ1}}\!\left(\mathrm{z1}(t)-\mathrm{z2}(t)+\frac{\theta 1(t)}{20}-\frac{4979\,\theta 2(t)}{100000}+\frac{3\,\phi 1(t)}{100}-\frac{2999\,\phi 2(t)}{100000}\right)+$$

$$\frac{3}{100}\,k_{\mathrm{pZ4}}\!\left(\mathrm{z1}(t)-\mathrm{z2}(t)-\frac{\theta 1(t)}{20}+\frac{4979\,\theta 2(t)}{100000}+\frac{3\,\phi 1(t)}{100}-\frac{2999\,\phi 2(t)}{100000}\right)-$$

$$\frac{3}{100}\,k_{\mathrm{pZ2}}\!\left(\mathrm{z1}(t)-\mathrm{z2}(t)+\frac{\theta 1(t)}{20}-\frac{4979\,\theta 2(t)}{100000}-\frac{3\,\phi 1(t)}{100}+\frac{2999\,\phi 2(t)}{100000}\right)-$$

$$\frac{3}{100}\,k_{\mathrm{pZ3}}\!\left(\mathrm{z1}(t)-\mathrm{z2}(t)-\frac{\theta 1(t)}{20}+\frac{4979\,\theta 2(t)}{100000}-\frac{3\,\phi 1(t)}{100}+\frac{2999\,\phi 2(t)}{100000}\right)+$$

$$\frac{1}{1000}\left(11\,k_{\mathrm{pX3}}\!\left(\mathrm{x1}(t)-\mathrm{x2}(t)+\frac{11\,\phi 1(t)}{1000}-\frac{347\,\phi 2(t)}{20000}+\frac{\psi 1(t)}{20}-\frac{4979\,\psi 2(t)}{100000}\right)\right)+$$

$$\frac{1}{1000}\left(11\,k_{\mathrm{pX4}}\!\left(\mathrm{x1}(t)-\mathrm{x2}(t)+\frac{11\,\phi 1(t)}{1000}-\frac{347\,\phi 2(t)}{20000}+\frac{\psi 1(t)}{20}-\frac{4979\,\psi 2(t)}{100000}\right)\right)+$$

$$\frac{1}{1000}\left(11\,k_{\mathrm{pX1}}\!\left(\mathrm{x1}(t)-\mathrm{x2}(t)+\frac{11\,\phi 1(t)}{1000}-\frac{347\,\phi 2(t)}{20000}-\frac{\psi 1(t)}{20}+\frac{4979\,\psi 2(t)}{100000}\right)\right)+$$

$$\frac{1}{1000}\left(11\,k_{\mathrm{pX2}}\!\left(\mathrm{x1}(t)-\mathrm{x2}(t)+\frac{11\,\phi 1(t)}{1000}-\frac{347\,\phi 2(t)}{20000}-\frac{\psi 1(t)}{20}+\frac{4979\,\psi 2(t)}{100000}\right)\right)-z_{\mathrm{in}}\,f_x(t)+$$

$$x_{\mathrm{in}}\,f_z(t)-m_y(t)+\frac{3}{100}\,b_{\mathrm{pZ1}}\!\left(\mathrm{z1}'(t)-\mathrm{z2}'(t)+\frac{\theta 1'(t)}{20}-\frac{4979\,\theta 2'(t)}{100000}+\frac{3\,\phi 1'(t)}{100}-\frac{2999\,\phi 2'(t)}{100000}\right)+$$

$$\frac{3}{100}\,b_{\mathrm{pZ4}}\!\left(\mathrm{z1}'(t)-\mathrm{z2}'(t)-\frac{\theta 1'(t)}{20}+\frac{4979\,\theta 2'(t)}{100000}+\frac{3\,\phi 1'(t)}{100}-\frac{2999\,\phi 2'(t)}{100000}\right)-$$

$$\frac{3}{100}\,b_{\mathrm{pZ2}}\!\left(\mathrm{z1}'(t)-\mathrm{z2}'(t)+\frac{\theta 1'(t)}{20}-\frac{4979\,\theta 2'(t)}{100000}-\frac{3\,\phi 1'(t)}{100}+\frac{2999\,\phi 2'(t)}{100000}\right)-$$

$$\frac{3}{100} b_{\text{pZ3}} \left(z1'(t) - z2'(t) - \frac{\theta 1'(t)}{20} + \frac{4979\,\theta 2'(t)}{100000} - \frac{3\,\phi 1'(t)}{100} + \frac{2999\,\phi 2'(t)}{100000} \right) +$$

$$\frac{1}{1000} \left(11 b_{\text{pX3}} \left(x1'(t) - x2'(t) + \frac{11\,\phi 1'(t)}{1000} - \frac{347\,\phi 2'(t)}{20000} + \frac{\psi 1'(t)}{20} - \frac{4979\,\psi 2'(t)}{100000} \right) \right) +$$

$$\frac{1}{1000} \left(11 b_{\text{pX4}} \left(x1'(t) - x2'(t) + \frac{11\,\phi 1'(t)}{1000} - \frac{347\,\phi 2'(t)}{20000} + \frac{\psi 1'(t)}{20} - \frac{4979\,\psi 2'(t)}{100000} \right) \right) +$$

$$\frac{1}{1000} \left(11 b_{\text{pX1}} \left(x1'(t) - x2'(t) + \frac{11\,\phi 1'(t)}{1000} - \frac{347\,\phi 2'(t)}{20000} - \frac{\psi 1'(t)}{20} + \frac{4979\,\psi 2'(t)}{100000} \right) \right) +$$

$$\frac{1}{1000} \left(11 b_{\text{pX2}} \left(x1''(t) - x2'(t) + \frac{11\,\phi 1'(t)}{1000} - \frac{347\,\phi 2'(t)}{20000} - \frac{\psi 1'(t)}{20} + \frac{4979\,\psi 2'(t)}{100000} \right) \right) + \text{JP}_{\text{yy}}\,\phi 1''(t) == 0,$$

$$\frac{1}{20} k_{\text{pX3}} \left(x1(t) - x2(t) + \frac{11\,\phi 1(t)}{1000} - \frac{347\,\phi 2(t)}{20000} + \frac{\psi 1(t)}{20} - \frac{4979\,\psi 2(t)}{100000} \right) +$$

$$\frac{1}{20} k_{\text{pX4}} \left(x1(t) - x2(t) + \frac{11\,\phi 1(t)}{1000} - \frac{347\,\phi 2(t)}{20000} + \frac{\psi 1(t)}{20} - \frac{4979\,\psi 2(t)}{100000} \right) +$$

$$\frac{1}{20} k_{\text{pX1}} \left(-x1(t) + x2(t) - \frac{11\,\phi 1(t)}{1000} + \frac{347\,\phi 2(t)}{20000} + \frac{\psi 1(t)}{20} - \frac{4979\,\psi 2(t)}{100000} \right) +$$

$$\frac{1}{20} k_{\text{pX2}} \left(-x1(t) + x2(t) - \frac{11\,\phi 1(t)}{1000} + \frac{347\,\phi 2(t)}{20000} + \frac{\psi 1(t)}{20} - \frac{4979\,\psi 2(t)}{100000} \right) +$$

$$\frac{3}{100} k_{\text{pY2}} \left(y1(t) - y2(t) - \frac{11\,\theta 1(t)}{1000} + \frac{347\,\theta 2(t)}{20000} + \frac{3\,\psi 1(t)}{100} - \frac{2999\,\psi 2(t)}{100000} \right) +$$

$$\frac{3}{100} k_{\text{pY3}} \left(y1(t) - y2(t) - \frac{11\,\theta 1(t)}{1000} + \frac{347\,\theta 2(t)}{20000} + \frac{3\,\psi 1(t)}{100} - \frac{2999\,\psi 2(t)}{100000} \right) -$$

$$\frac{3}{100} k_{\text{pY1}} \left(y1(t) - y2(t) - \frac{11\,\theta 1(t)}{1000} + \frac{347\,\theta 2(t)}{20000} - \frac{3\,\psi 1(t)}{100} + \frac{2999\,\psi 2(t)}{100000} \right) -$$

$$\frac{3}{100} k_{\text{pY4}} \left(y1(t) - y2(t) - \frac{11\,\theta 1(t)}{1000} + \frac{347\,\theta 2(t)}{20000} - \frac{3\,\psi 1(t)}{100} + \frac{2999\,\psi 2(t)}{100000} \right) + y_{\text{in}}\,f_x(t) -$$

$$x_{\text{in}}\,f_y(t) - m_z(t) + \frac{1}{20} b_{\text{pX3}} \left(x1'(t) - x2'(t) + \frac{11\,\phi 1'(t)}{1000} - \frac{347\,\phi 2'(t)}{20000} + \frac{\psi 1'(t)}{20} - \frac{4979\,\psi 2'(t)}{100000} \right) +$$

$$\frac{1}{20} b_{\text{pX4}} \left(x1'(t) - x2'(t) + \frac{11\,\phi 1'(t)}{1000} - \frac{347\,\phi 2'(t)}{20000} + \frac{\psi 1'(t)}{20} - \frac{4979\,\psi 2'(t)}{100000} \right) +$$

$$\frac{3}{100} b_{\text{pY2}} \left(y1'(t) - y2'(t) - \frac{11\,\theta 1'(t)}{1000} + \frac{347\,\theta 2'(t)}{20000} + \frac{3\,\psi 1'(t)}{100} - \frac{2999\,\psi 2'(t)}{100000} \right) +$$

$$\frac{3}{100} b_{\text{pY3}} \left(y1'(t) - y2'(t) - \frac{11\,\theta 1'(t)}{1000} + \frac{347\,\theta 2'(t)}{20000} + \frac{3\,\psi 1'(t)}{100} - \frac{2999\,\psi 2'(t)}{100000} \right) -$$

$$\frac{3}{100} b_{\text{pY1}} \left(y1'(t) - y2'(t) - \frac{11\,\theta 1'(t)}{1000} + \frac{347\,\theta 2'(t)}{20000} - \frac{3\,\psi 1'(t)}{100} + \frac{2999\,\psi 2'(t)}{100000} \right) -$$

$$\frac{3}{100} b_{\text{pY4}} \left(y1'(t) - y2'(t) - \frac{11\,\theta 1'(t)}{1000} + \frac{347\,\theta 2'(t)}{20000} - \frac{3\,\psi 1'(t)}{100} + \frac{2999\,\psi 2'(t)}{100000} \right) -$$

$$\frac{1}{20} b_{\text{pX1}} \left(x1'(t) - x2'(t) + \frac{11\,\phi 1'(t)}{1000} - \frac{347\,\phi 2'(t)}{20000} - \frac{\psi 1'(t)}{20} + \frac{4979\,\psi 2'(t)}{100000} \right) -$$

$$\frac{1}{20} b_{\text{pX2}} \left(x1'(t) - x2'(t) + \frac{11\,\phi 1'(t)}{1000} - \frac{347\,\phi 2'(t)}{20000} - \frac{\psi 1'(t)}{20} + \frac{4979\,\psi 2'(t)}{100000} \right) + \text{JP}_{\text{zz}}\,\psi 1''(t) == 0,$$

$$k_{\text{bX1}} \left(x2(t) + \frac{1467\,\phi 2(t)}{20000} - \frac{3\,\psi 2(t)}{20} \right) + k_{\text{bX2}} \left(x2(t) + \frac{1467\,\phi 2(t)}{20000} - \frac{3\,\psi 2(t)}{20} \right) -$$

$$k_{\text{pX3}} \left(x1(t) - x2(t) + \frac{11\,\phi 1(t)}{1000} - \frac{347\,\phi 2(t)}{20000} + \frac{\psi 1(t)}{20} - \frac{4979\,\psi 2(t)}{100000} \right) -$$

$$k_{\mathrm{pX4}}\left(\mathrm{x1}(t) - \mathrm{x2}(t) + \frac{11\,\phi1(t)}{1000} - \frac{347\,\phi2(t)}{20000} + \frac{\psi1(t)}{20} - \frac{4979\,\psi2(t)}{100000}\right) -$$

$$k_{\mathrm{pX1}}\left(\mathrm{x1}(t) - \mathrm{x2}(t) + \frac{11\,\phi1(t)}{1000} - \frac{347\,\phi2(t)}{20000} - \frac{\psi1(t)}{20} + \frac{4979\,\psi2(t)}{100000}\right) -$$

$$k_{\mathrm{pX2}}\left(\mathrm{x1}(t) - \mathrm{x2}(t) + \frac{11\,\phi1(t)}{1000} - \frac{347\,\phi2(t)}{20000} - \frac{\psi1(t)}{20} + \frac{4979\,\psi2(t)}{100000}\right) +$$

$$k_{\mathrm{bX3}}\left(\mathrm{x2}(t) + \frac{1467\,\phi2(t)}{20000} + \frac{3\,\psi2(t)}{20}\right) + k_{\mathrm{bX4}}\left(\mathrm{x2}(t) + \frac{1467\,\phi2(t)}{20000} + \frac{3\,\psi2(t)}{20}\right) - f_{\mathrm{xB}}(t) +$$

$$b_{\mathrm{bX1}}\left(\mathrm{x2}'(t) + \frac{1467\,\phi2'(t)}{20000} - \frac{3\,\psi2'(t)}{20}\right) + b_{\mathrm{bX2}}\left(\mathrm{x2}'(t) + \frac{1467\,\phi2'(t)}{20000} - \frac{3\,\psi2'(t)}{20}\right) -$$

$$b_{\mathrm{pX3}}\left(\mathrm{x1}'(t) - \mathrm{x2}'(t) + \frac{11\,\phi1'(t)}{1000} - \frac{347\,\phi2'(t)}{20000} + \frac{\psi1'(t)}{20} - \frac{4979\,\psi2'(t)}{100000}\right) -$$

$$b_{\mathrm{pX4}}\left(\mathrm{x1}'(t) - \mathrm{x2}'(t) + \frac{11\,\phi1'(t)}{1000} - \frac{347\,\phi2'(t)}{20000} + \frac{\psi1'(t)}{20} - \frac{4979\,\psi2'(t)}{100000}\right) -$$

$$b_{\mathrm{pX1}}\left(\mathrm{x1}'(t) - \mathrm{x2}'(t) + \frac{11\,\phi1'(t)}{1000} - \frac{347\,\phi2'(t)}{20000} - \frac{\psi1'(t)}{20} + \frac{4979\,\psi2'(t)}{100000}\right) -$$

$$b_{\mathrm{pX2}}\left(\mathrm{x1}'(t) - \mathrm{x2}'(t) + \frac{11\,\phi1'(t)}{1000} - \frac{347\,\phi2'(t)}{20000} - \frac{\psi1'(t)}{20} + \frac{4979\,\psi2'(t)}{100000}\right) +$$

$$b_{\mathrm{bX3}}\left(\mathrm{x2}'(t) + \frac{1467\,\phi2'(t)}{20000} + \frac{3\,\psi2'(t)}{20}\right) + b_{\mathrm{bX4}}\left(\mathrm{x2}'(t) + \frac{1467\,\phi2'(t)}{20000} + \frac{3\,\psi2'(t)}{20}\right) + \mathrm{mB}\,\mathrm{x2}''(t) == 0,$$

$$k_{\mathrm{bY1}}\left(\mathrm{y2}(t) - \frac{1467\,\theta2(t)}{20000} - \frac{3\,\psi2(t)}{10}\right) + k_{\mathrm{bY4}}\left(\mathrm{y2}(t) - \frac{1467\,\theta2(t)}{20000} - \frac{3\,\psi2(t)}{10}\right) -$$

$$k_{\mathrm{pY2}}\left(\mathrm{y1}(t) - \mathrm{y2}(t) - \frac{11\,\theta1(t)}{1000} + \frac{347\,\theta2(t)}{20000} + \frac{3\,\psi1(t)}{100} - \frac{2999\,\psi2(t)}{100000}\right) -$$

$$k_{\mathrm{pY3}}\left(\mathrm{y1}(t) - \mathrm{y2}(t) - \frac{11\,\theta1(t)}{1000} + \frac{347\,\theta2(t)}{20000} + \frac{3\,\psi1(t)}{100} - \frac{2999\,\psi2(t)}{100000}\right) -$$

$$k_{\mathrm{pY1}}\left(\mathrm{y1}(t) - \mathrm{y2}(t) - \frac{11\,\theta1(t)}{1000} + \frac{347\,\theta2(t)}{20000} - \frac{3\,\psi1(t)}{100} + \frac{2999\,\psi2(t)}{100000}\right) -$$

$$k_{\mathrm{pY4}}\left(\mathrm{y1}(t) - \mathrm{y2}(t) - \frac{11\,\theta1(t)}{1000} + \frac{347\,\theta2(t)}{20000} - \frac{3\,\psi1(t)}{100} + \frac{2999\,\psi2(t)}{100000}\right) +$$

$$k_{\mathrm{bY2}}\left(\mathrm{y2}(t) - \frac{1467\,\theta2(t)}{20000} + \frac{3\,\psi2(t)}{10}\right) + k_{\mathrm{bY3}}\left(\mathrm{y2}(t) - \frac{1467\,\theta2(t)}{20000} + \frac{3\,\psi2(t)}{10}\right) - f_{\mathrm{yB}}(t) +$$

$$b_{\mathrm{bY1}}\left(\mathrm{y2}'(t) - \frac{1467\,\theta2'(t)}{20000} - \frac{3\,\psi2'(t)}{10}\right) + b_{\mathrm{bY4}}\left(\mathrm{y2}'(t) - \frac{1467\,\theta2'(t)}{20000} - \frac{3\,\psi2'(t)}{10}\right) -$$

$$b_{\mathrm{pY2}}\left(\mathrm{y1}'(t) - \mathrm{y2}'(t) - \frac{11\,\theta1'(t)}{1000} + \frac{347\,\theta2'(t)}{20000} + \frac{3\,\psi1'(t)}{100} - \frac{2999\,\psi2'(t)}{100000}\right) -$$

$$b_{\mathrm{pY3}}\left(\mathrm{y1}'(t) - \mathrm{y2}'(t) - \frac{11\,\theta1'(t)}{1000} + \frac{347\,\theta2'(t)}{20000} + \frac{3\,\psi1'(t)}{100} - \frac{2999\,\psi2'(t)}{100000}\right) -$$

$$b_{\mathrm{pY1}}\left(\mathrm{y1}'(t) - \mathrm{y2}'(t) - \frac{11\,\theta1'(t)}{1000} + \frac{347\,\theta2'(t)}{20000} - \frac{3\,\psi1'(t)}{100} + \frac{2999\,\psi2'(t)}{100000}\right) -$$

$$b_{\mathrm{pY4}}\left(\mathrm{y1}'(t) - \mathrm{y2}'(t) - \frac{11\,\theta1'(t)}{1000} + \frac{347\,\theta2'(t)}{20000} - \frac{3\,\psi1'(t)}{100} + \frac{2999\,\psi2'(t)}{100000}\right) +$$

$$b_{\mathrm{bY2}}\left(\mathrm{y2}'(t) - \frac{1467\,\theta2'(t)}{20000} + \frac{3\,\psi2'(t)}{10}\right) + b_{\mathrm{bY3}}\left(\mathrm{y2}'(t) - \frac{1467\,\theta2'(t)}{20000} + \frac{3\,\psi2'(t)}{10}\right) + \mathrm{mB}\,\mathrm{y2}''(t) == 0,$$

$$\mathrm{z2}''(t)\,\mathrm{mB} + \frac{981\,\mathrm{mB}}{100} + k_{\mathrm{bZ3}}\left(\mathrm{z2}(t) - \frac{3\,\theta2(t)}{20} - \frac{3\,\phi2(t)}{10}\right) + k_{\mathrm{bZ2}}\left(\mathrm{z2}(t) + \frac{3\,\theta2(t)}{20} - \frac{3\,\phi2(t)}{10}\right) -$$

$$k_{\mathrm{pZ1}}\left(z1(t) - z2(t) + \frac{\theta 1(t)}{20} - \frac{4979\,\theta 2(t)}{100000} + \frac{3\,\phi 1(t)}{100} - \frac{2999\,\phi 2(t)}{100000}\right) -$$

$$k_{\mathrm{pZ4}}\left(z1(t) - z2(t) - \frac{\theta 1(t)}{20} + \frac{4979\,\theta 2(t)}{100000} + \frac{3\,\phi 1(t)}{100} - \frac{2999\,\phi 2(t)}{100000}\right) -$$

$$k_{\mathrm{pZ2}}\left(z1(t) - z2(t) + \frac{\theta 1(t)}{20} - \frac{4979\,\theta 2(t)}{100000} - \frac{3\,\phi 1(t)}{100} + \frac{2999\,\phi 2(t)}{100000}\right) -$$

$$k_{\mathrm{pZ3}}\left(z1(t) - z2(t) - \frac{\theta 1(t)}{20} + \frac{4979\,\theta 2(t)}{100000} - \frac{3\,\phi 1(t)}{100} + \frac{2999\,\phi 2(t)}{100000}\right) + k_{\mathrm{bZ4}}\left(z2(t) - \frac{3\,\theta 2(t)}{20} + \frac{3\,\phi 2(t)}{10}\right) +$$

$$k_{\mathrm{bZ1}}\left(z2(t) + \frac{3\,\theta 2(t)}{20} + \frac{3\,\phi 2(t)}{10}\right) - f_{\mathrm{zB}}(t) + b_{\mathrm{bZ3}}\left(z2'(t) - \frac{3\,\theta 2'(t)}{20} - \frac{3\,\phi 2'(t)}{10}\right) +$$

$$b_{\mathrm{bZ2}}\left(z2'(t) + \frac{3\,\theta 2'(t)}{20} - \frac{3\,\phi 2'(t)}{10}\right) - b_{\mathrm{pZ1}}\left(z1'(t) - z2'(t) + \frac{\theta 1'(t)}{20} - \frac{4979\,\theta 2'(t)}{100000} + \frac{3\,\phi 1'(t)}{100} - \frac{2999\,\phi 2'(t)}{100000}\right) -$$

$$b_{\mathrm{pZ4}}\left(z1'(t) - z2'(t) - \frac{\theta 1'(t)}{20} + \frac{4979\,\theta 2'(t)}{100000} + \frac{3\,\phi 1'(t)}{100} - \frac{2999\,\phi 2'(t)}{100000}\right) -$$

$$b_{\mathrm{pZ2}}\left(z1'(t) - z2'(t) + \frac{\theta 1'(t)}{20} - \frac{4979\,\theta 2'(t)}{100000} - \frac{3\,\phi 1'(t)}{100} + \frac{2999\,\phi 2'(t)}{100000}\right) -$$

$$b_{\mathrm{pZ3}}\left(z1'(t) - z2'(t) - \frac{\theta 1'(t)}{20} + \frac{4979\,\theta 2'(t)}{100000} - \frac{3\,\phi 1'(t)}{100} + \frac{2999\,\phi 2'(t)}{100000}\right) +$$

$$b_{\mathrm{bZ4}}\left(z2'(t) - \frac{3\,\theta 2'(t)}{20} + \frac{3\,\phi 2'(t)}{10}\right) + b_{\mathrm{bZ1}}\left(z2'(t) + \frac{3\,\theta 2'(t)}{20} + \frac{3\,\phi 2'(t)}{10}\right) == 0,$$

$$-\frac{3}{20}k_{\mathrm{bZ3}}\left(z2(t) - \frac{3\,\theta 2(t)}{20} - \frac{3\,\phi 2(t)}{10}\right) + \frac{3}{20}k_{\mathrm{bZ2}}\left(z2(t) + \frac{3\,\theta 2(t)}{20} - \frac{3\,\phi 2(t)}{10}\right) -$$

$$\frac{1}{100000}\left(4979\,k_{\mathrm{pZ1}}\left(z1(t) - z2(t) + \frac{\theta 1(t)}{20} - \frac{4979\,\theta 2(t)}{100000} + \frac{3\,\phi 1(t)}{100} - \frac{2999\,\phi 2(t)}{100000}\right)\right) +$$

$$\frac{1}{100000}\left(4979\,k_{\mathrm{pZ4}}\left(z1(t) - z2(t) - \frac{\theta 1(t)}{20} + \frac{4979\,\theta 2(t)}{100000} + \frac{3\,\phi 1(t)}{100} - \frac{2999\,\phi 2(t)}{100000}\right)\right) -$$

$$\frac{1}{100000}\left(4979\,k_{\mathrm{pZ2}}\left(z1(t) - z2(t) + \frac{\theta 1(t)}{20} - \frac{4979\,\theta 2(t)}{100000} - \frac{3\,\phi 1(t)}{100} + \frac{2999\,\phi 2(t)}{100000}\right)\right) +$$

$$\frac{1}{100000}\left(4979\,k_{\mathrm{pZ3}}\left(z1(t) - z2(t) - \frac{\theta 1(t)}{20} + \frac{4979\,\theta 2(t)}{100000} - \frac{3\,\phi 1(t)}{100} + \frac{2999\,\phi 2(t)}{100000}\right)\right) -$$

$$\frac{3}{20}k_{\mathrm{bZ4}}\left(z2(t) - \frac{3\,\theta 2(t)}{20} + \frac{3\,\phi 2(t)}{10}\right) + \frac{3}{20}k_{\mathrm{bZ1}}\left(z2(t) + \frac{3\,\theta 2(t)}{20} + \frac{3\,\phi 2(t)}{10}\right) -$$

$$\frac{1467\,k_{\mathrm{bY1}}\left(y2(t) - \frac{1467\,\theta 2(t)}{20000} - \frac{3\,\psi 2(t)}{10}\right)}{20000} - \frac{1467\,k_{\mathrm{bY4}}\left(y2(t) - \frac{1467\,\theta 2(t)}{20000} - \frac{3\,\psi 2(t)}{10}\right)}{20000} +$$

$$\frac{1}{20000}\left(347\,k_{\mathrm{pY2}}\left(y1(t) - y2(t) - \frac{11\,\theta 1(t)}{1000} + \frac{347\,\theta 2(t)}{20000} + \frac{3\,\psi 1(t)}{100} - \frac{2999\,\psi 2(t)}{100000}\right)\right) +$$

$$\frac{1}{20000}\left(347\,k_{\mathrm{pY3}}\left(y1(t) - y2(t) - \frac{11\,\theta 1(t)}{1000} + \frac{347\,\theta 2(t)}{20000} + \frac{3\,\psi 1(t)}{100} - \frac{2999\,\psi 2(t)}{100000}\right)\right) +$$

$$\frac{1}{20000}\left(347\,k_{\mathrm{pY1}}\left(y1(t) - y2(t) - \frac{11\,\theta 1(t)}{1000} + \frac{347\,\theta 2(t)}{20000} - \frac{3\,\psi 1(t)}{100} + \frac{2999\,\psi 2(t)}{100000}\right)\right) +$$

$$\frac{1}{20000}\left(347\,k_{\mathrm{pY4}}\left(y1(t) - y2(t) - \frac{11\,\theta 1(t)}{1000} + \frac{347\,\theta 2(t)}{20000} - \frac{3\,\psi 1(t)}{100} + \frac{2999\,\psi 2(t)}{100000}\right)\right) -$$

$$\frac{1467\,k_{\mathrm{bY2}}\left(y2(t) - \frac{1467\,\theta 2(t)}{20000} + \frac{3\,\psi 2(t)}{10}\right)}{20000} -$$

$$\frac{1467\,k_{\mathrm{bY3}}\left(\mathrm{y2}(t) - \frac{1467\,\theta2(t)}{20000} + \frac{3\,\psi2(t)}{10}\right)}{20000} + z_{\mathrm{inb}}\,f_{\mathrm{yB}}(t) - y_{\mathrm{inb}}\,f_{\mathrm{zB}}(t) - m_{\mathrm{xB}}(t) -$$

$$\frac{3}{20}\,b_{\mathrm{bZ3}}\left(\mathrm{z2}'(t) - \frac{3\,\theta2'(t)}{20} - \frac{3\,\phi2'(t)}{10}\right) + \frac{3}{20}\,b_{\mathrm{bZ2}}\left(\mathrm{z2}'(t) + \frac{3\,\theta2'(t)}{20} - \frac{3\,\phi2'(t)}{10}\right) -$$

$$\frac{1}{100000}\left(4979\,b_{\mathrm{pZ1}}\left(\mathrm{z1}'(t) - \mathrm{z2}'(t) + \frac{\theta1'(t)}{20} - \frac{4979\,\theta2'(t)}{100000} + \frac{3\,\phi1'(t)}{100} - \frac{2999\,\phi2'(t)}{100000}\right)\right) +$$

$$\frac{1}{100000}\left(4979\,b_{\mathrm{pZ4}}\left(\mathrm{z1}'(t) - \mathrm{z2}'(t) - \frac{\theta1'(t)}{20} + \frac{4979\,\theta2'(t)}{100000} + \frac{3\,\phi1'(t)}{100} - \frac{2999\,\phi2'(t)}{100000}\right)\right) -$$

$$\frac{1}{100000}\left(4979\,b_{\mathrm{pZ2}}\left(\mathrm{z1}'(t) - \mathrm{z2}'(t) + \frac{\theta1'(t)}{20} - \frac{4979\,\theta2'(t)}{100000} - \frac{3\,\phi1'(t)}{100} + \frac{2999\,\phi2'(t)}{100000}\right)\right) +$$

$$\frac{1}{100000}\left(4979\,b_{\mathrm{pZ3}}\left(\mathrm{z1}'(t) - \mathrm{z2}'(t) - \frac{\theta1'(t)}{20} + \frac{4979\,\theta2'(t)}{100000} - \frac{3\,\phi1'(t)}{100} + \frac{2999\,\phi2'(t)}{100000}\right)\right) -$$

$$\frac{3}{20}\,b_{\mathrm{bZ4}}\left(\mathrm{z2}'(t) - \frac{3\,\theta2'(t)}{20} + \frac{3\,\phi2'(t)}{10}\right) + \frac{3}{20}\,b_{\mathrm{bZ1}}\left(\mathrm{z2}'(t) + \frac{3\,\theta2'(t)}{20} + \frac{3\,\phi2'(t)}{10}\right) -$$

$$\frac{1467\,b_{\mathrm{bY1}}\left(\mathrm{y2}'(t) - \frac{1467\,\theta2'(t)}{20000} - \frac{3\,\psi2'(t)}{10}\right)}{20000} - \frac{1467\,b_{\mathrm{bY4}}\left(\mathrm{y2}'(t) - \frac{1467\,\theta2'(t)}{20000} - \frac{3\,\psi2'(t)}{10}\right)}{20000} +$$

$$\frac{1}{20000}\left(347\,b_{\mathrm{pY2}}\left(\mathrm{y1}'(t) - \mathrm{y2}'(t) - \frac{11\,\theta1'(t)}{1000} + \frac{347\,\theta2'(t)}{20000} + \frac{3\,\psi1'(t)}{100} - \frac{2999\,\psi2'(t)}{100000}\right)\right) +$$

$$\frac{1}{20000}\left(347\,b_{\mathrm{pY3}}\left(\mathrm{y1}'(t) - \mathrm{y2}'(t) - \frac{11\,\theta1'(t)}{1000} + \frac{347\,\theta2'(t)}{20000} + \frac{3\,\psi1'(t)}{100} - \frac{2999\,\psi2'(t)}{100000}\right)\right) +$$

$$\frac{1}{20000}\left(347\,b_{\mathrm{pY1}}\left(\mathrm{y1}'(t) - \mathrm{y2}'(t) - \frac{11\,\theta1'(t)}{1000} + \frac{347\,\theta2'(t)}{20000} - \frac{3\,\psi1'(t)}{100} + \frac{2999\,\psi2'(t)}{100000}\right)\right) +$$

$$\frac{1}{20000}\left(347\,b_{\mathrm{pY4}}\left(\mathrm{y1}'(t) - \mathrm{y2}'(t) - \frac{11\,\theta1'(t)}{1000} + \frac{347\,\theta2'(t)}{20000} - \frac{3\,\psi1'(t)}{100} + \frac{2999\,\psi2'(t)}{100000}\right)\right) -$$

$$\frac{1467\,b_{\mathrm{bY2}}\left(\mathrm{y2}'(t) - \frac{1467\,\theta2'(t)}{20000} + \frac{3\,\psi2'(t)}{10}\right)}{20000} - \frac{1467\,b_{\mathrm{bY3}}\left(\mathrm{y2}'(t) - \frac{1467\,\theta2'(t)}{20000} + \frac{3\,\psi2'(t)}{10}\right)}{20000} + \mathrm{JB}_{\mathrm{xx}}\,\theta2''(t) == 0,$$

$$-\frac{3}{10}\,k_{\mathrm{bZ3}}\left(\mathrm{z2}(t) - \frac{3\,\theta2(t)}{20} - \frac{3\,\phi2(t)}{10}\right) - \frac{3}{10}\,k_{\mathrm{bZ2}}\left(\mathrm{z2}(t) + \frac{3\,\theta2(t)}{20} - \frac{3\,\phi2(t)}{10}\right) -$$

$$\frac{1}{100000}\left(2999\,k_{\mathrm{pZ1}}\left(\mathrm{z1}(t) - \mathrm{z2}(t) + \frac{\theta1(t)}{20} - \frac{4979\,\theta2(t)}{100000} + \frac{3\,\phi1(t)}{100} - \frac{2999\,\phi2(t)}{100000}\right)\right) -$$

$$\frac{1}{100000}\left(2999\,k_{\mathrm{pZ4}}\left(\mathrm{z1}(t) - \mathrm{z2}(t) - \frac{\theta1(t)}{20} + \frac{4979\,\theta2(t)}{100000} + \frac{3\,\phi1(t)}{100} - \frac{2999\,\phi2(t)}{100000}\right)\right) +$$

$$\frac{1}{100000}\left(2999\,k_{\mathrm{pZ2}}\left(\mathrm{z1}(t) - \mathrm{z2}(t) + \frac{\theta1(t)}{20} - \frac{4979\,\theta2(t)}{100000} - \frac{3\,\phi1(t)}{100} + \frac{2999\,\phi2(t)}{100000}\right)\right) +$$

$$\frac{1}{100000}\left(2999\,k_{\mathrm{pZ3}}\left(\mathrm{z1}(t) - \mathrm{z2}(t) - \frac{\theta1(t)}{20} + \frac{4979\,\theta2(t)}{100000} - \frac{3\,\phi1(t)}{100} + \frac{2999\,\phi2(t)}{100000}\right)\right) +$$

$$\frac{3}{10}\,k_{\mathrm{bZ4}}\left(\mathrm{z2}(t) - \frac{3\,\theta2(t)}{20} + \frac{3\,\phi2(t)}{10}\right) + \frac{3}{10}\,k_{\mathrm{bZ1}}\left(\mathrm{z2}(t) + \frac{3\,\theta2(t)}{20} + \frac{3\,\phi2(t)}{10}\right) +$$

$$\frac{1467\,k_{\mathrm{bX1}}\left(\mathrm{x2}(t) + \frac{1467\,\phi2(t)}{20000} - \frac{3\,\psi2(t)}{20}\right)}{20000} + \frac{1467\,k_{\mathrm{bX2}}\left(\mathrm{x2}(t) + \frac{1467\,\phi2(t)}{20000} - \frac{3\,\psi2(t)}{20}\right)}{20000} -$$

$$\frac{1}{20000}\left(347\,k_{\mathrm{pX3}}\left(\mathrm{x1}(t) - \mathrm{x2}(t) + \frac{11\,\phi1(t)}{1000} - \frac{347\,\phi2(t)}{20000} + \frac{\psi1(t)}{20} - \frac{4979\,\psi2(t)}{100000}\right)\right) -$$

$$\frac{1}{20000}\left(347\,k_{\mathrm{pX4}}\left(\mathrm{x1}(t) - \mathrm{x2}(t) + \frac{11\,\phi1(t)}{1000} - \frac{347\,\phi2(t)}{20000} + \frac{\psi1(t)}{20} - \frac{4979\,\psi2(t)}{100000}\right)\right) -$$

$$\frac{1}{20000}\left(347\,k_{\text{pX1}}\left(\text{x1}(t)-\text{x2}(t)+\frac{11\,\phi1(t)}{1000}-\frac{347\,\phi2(t)}{20000}-\frac{\psi1(t)}{20}+\frac{4979\,\psi2(t)}{100000}\right)\right)-$$

$$\frac{1}{20000}\left(347\,k_{\text{pX2}}\left(\text{x1}(t)-\text{x2}(t)+\frac{11\,\phi1(t)}{1000}-\frac{347\,\phi2(t)}{20000}-\frac{\psi1(t)}{20}+\frac{4979\,\psi2(t)}{100000}\right)\right)+$$

$$\frac{1467\,k_{\text{bX3}}\left(\text{x2}(t)+\frac{1467\,\phi2(t)}{20000}+\frac{3\,\psi2(t)}{20}\right)}{20000}+\frac{1467\,k_{\text{bX4}}\left(\text{x2}(t)+\frac{1467\,\phi2(t)}{20000}+\frac{3\,\psi2(t)}{20}\right)}{20000}-z_{\text{inb}}f_{\text{xB}}(t)+$$

$$x_{\text{inb}}f_{\text{zB}}(t)-m_{\text{yB}}(t)-\frac{3}{10}\,b_{\text{bZ3}}\left(\text{z2}'(t)-\frac{3\,\theta2'(t)}{20}-\frac{3\,\phi2'(t)}{10}\right)-\frac{3}{10}\,b_{\text{bZ2}}\left(\text{z2}'(t)+\frac{3\,\theta2'(t)}{20}-\frac{3\,\phi2'(t)}{10}\right)-$$

$$\frac{1}{100000}\left(2999\,b_{\text{pZ1}}\left(\text{z1}'(t)-\text{z2}'(t)+\frac{\theta1'(t)}{20}-\frac{4979\,\theta2'(t)}{100000}+\frac{3\,\phi1'(t)}{100}-\frac{2999\,\phi2'(t)}{100000}\right)\right)-$$

$$\frac{1}{100000}\left(2999\,b_{\text{pZ4}}\left(\text{z1}'(t)-\text{z2}'(t)-\frac{\theta1'(t)}{20}+\frac{4979\,\theta2'(t)}{100000}+\frac{3\,\phi1'(t)}{100}-\frac{2999\,\phi2'(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(2999\,b_{\text{pZ2}}\left(\text{z1}'(t)-\text{z2}'(t)+\frac{\theta1'(t)}{20}-\frac{4979\,\theta2'(t)}{100000}-\frac{3\,\phi1'(t)}{100}+\frac{2999\,\phi2'(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(2999\,b_{\text{pZ3}}\left(\text{z1}'(t)-\text{z2}'(t)-\frac{\theta1'(t)}{20}+\frac{4979\,\theta2'(t)}{100000}-\frac{3\,\phi1'(t)}{100}+\frac{2999\,\phi2'(t)}{100000}\right)\right)+$$

$$\frac{3}{10}\,b_{\text{bZ4}}\left(\text{z2}'(t)-\frac{3\,\theta2'(t)}{20}+\frac{3\,\phi2'(t)}{10}\right)+\frac{3}{10}\,b_{\text{bZ1}}\left(\text{z2}'(t)+\frac{3\,\theta2'(t)}{20}+\frac{3\,\phi2'(t)}{10}\right)+$$

$$\frac{1467\,b_{\text{bX1}}\left(\text{x2}'(t)+\frac{1467\,\phi2'(t)}{20000}-\frac{3\,\psi2'(t)}{20}\right)}{20000}+\frac{1467\,b_{\text{bX2}}\left(\text{x2}'(t)+\frac{1467\,\phi2'(t)}{20000}-\frac{3\,\psi2'(t)}{20}\right)}{20000}-$$

$$\frac{1}{20000}\left(347\,b_{\text{pX3}}\left(\text{x1}'(t)-\text{x2}'(t)+\frac{11\,\phi1'(t)}{1000}-\frac{347\,\phi2'(t)}{20000}+\frac{\psi1'(t)}{20}-\frac{4979\,\psi2'(t)}{100000}\right)\right)-$$

$$\frac{1}{20000}\left(347\,b_{\text{pX4}}\left(\text{x1}'(t)-\text{x2}'(t)+\frac{11\,\phi1'(t)}{1000}-\frac{347\,\phi2'(t)}{20000}+\frac{\psi1'(t)}{20}-\frac{4979\,\psi2'(t)}{100000}\right)\right)-$$

$$\frac{1}{20000}\left(347\,b_{\text{pX1}}\left(\text{x1}'(t)-\text{x2}'(t)+\frac{11\,\phi1'(t)}{1000}-\frac{347\,\phi2'(t)}{20000}-\frac{\psi1'(t)}{20}+\frac{4979\,\psi2'(t)}{100000}\right)\right)-$$

$$\frac{1}{20000}\left(347\,b_{\text{pX2}}\left(\text{x1}'(t)-\text{x2}'(t)+\frac{11\,\phi1'(t)}{1000}-\frac{347\,\phi2'(t)}{20000}-\frac{\psi1'(t)}{20}+\frac{4979\,\psi2'(t)}{100000}\right)\right)+$$

$$\frac{1467\,b_{\text{bX3}}\left(\text{x2}'(t)+\frac{1467\,\phi2'(t)}{20000}+\frac{3\,\psi2'(t)}{20}\right)}{20000}+\frac{1467\,b_{\text{bX4}}\left(\text{x2}'(t)+\frac{1467\,\phi2'(t)}{20000}+\frac{3\,\psi2'(t)}{20}\right)}{20000}+JB_{\text{yy}}\,\phi2''(t)==0,$$

$$-\frac{3}{10}\,k_{\text{bY1}}\left(\text{y2}(t)-\frac{1467\,\theta2(t)}{20000}-\frac{3\,\psi2(t)}{10}\right)-\frac{3}{10}\,k_{\text{bY4}}\left(\text{y2}(t)-\frac{1467\,\theta2(t)}{20000}-\frac{3\,\psi2(t)}{10}\right)-$$

$$\frac{3}{20}\,k_{\text{bX1}}\left(\text{x2}(t)+\frac{1467\,\phi2(t)}{20000}-\frac{3\,\psi2(t)}{20}\right)-\frac{3}{20}\,k_{\text{bX2}}\left(\text{x2}(t)+\frac{1467\,\phi2(t)}{20000}-\frac{3\,\psi2(t)}{20}\right)-$$

$$\frac{1}{100000}\left(4979\,k_{\text{pX3}}\left(\text{x1}(t)-\text{x2}(t)+\frac{11\,\phi1(t)}{1000}-\frac{347\,\phi2(t)}{20000}+\frac{\psi1(t)}{20}-\frac{4979\,\psi2(t)}{100000}\right)\right)-$$

$$\frac{1}{100000}\left(4979\,k_{\text{pX4}}\left(\text{x1}(t)-\text{x2}(t)+\frac{11\,\phi1(t)}{1000}-\frac{347\,\phi2(t)}{20000}+\frac{\psi1(t)}{20}-\frac{4979\,\psi2(t)}{100000}\right)\right)-$$

$$\frac{1}{100000}\left(2999\,k_{\text{pY2}}\left(\text{y1}(t)-\text{y2}(t)-\frac{11\,\theta1(t)}{1000}+\frac{347\,\theta2(t)}{20000}+\frac{3\,\psi1(t)}{100}-\frac{2999\,\psi2(t)}{100000}\right)\right)-$$

$$\frac{1}{100000}\left(2999\,k_{\text{pY3}}\left(\text{y1}(t)-\text{y2}(t)-\frac{11\,\theta1(t)}{1000}+\frac{347\,\theta2(t)}{20000}+\frac{3\,\psi1(t)}{100}-\frac{2999\,\psi2(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(2999\,k_{\text{pY1}}\left(\text{y1}(t)-\text{y2}(t)-\frac{11\,\theta1(t)}{1000}+\frac{347\,\theta2(t)}{20000}-\frac{3\,\psi1(t)}{100}+\frac{2999\,\psi2(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(2999\,k_{\text{pY4}}\left(\text{y1}(t)-\text{y2}(t)-\frac{11\,\theta 1(t)}{1000}+\frac{347\,\theta 2(t)}{20000}-\frac{3\,\psi 1(t)}{100}+\frac{2999\,\psi 2(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(4979\,k_{\text{pX1}}\left(\text{x1}(t)-\text{x2}(t)+\frac{11\,\phi 1(t)}{1000}-\frac{347\,\phi 2(t)}{20000}-\frac{\psi 1(t)}{20}+\frac{4979\,\psi 2(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(4979\,k_{\text{pX2}}\left(\text{x1}(t)-\text{x2}(t)+\frac{11\,\phi 1(t)}{1000}-\frac{347\,\phi 2(t)}{20000}-\frac{\psi 1(t)}{20}+\frac{4979\,\psi 2(t)}{100000}\right)\right)+$$

$$\frac{3}{20}\,k_{\text{bX3}}\left(\text{x2}(t)+\frac{1467\,\phi 2(t)}{20000}+\frac{3\,\psi 2(t)}{20}\right)+\frac{3}{20}\,k_{\text{bX4}}\left(\text{x2}(t)+\frac{1467\,\phi 2(t)}{20000}+\frac{3\,\psi 2(t)}{20}\right)+$$

$$\frac{3}{10}\,k_{\text{bY2}}\left(\text{y2}(t)-\frac{1467\,\theta 2(t)}{20000}+\frac{3\,\psi 2(t)}{10}\right)+\frac{3}{10}\,k_{\text{bY3}}\left(\text{y2}(t)-\frac{1467\,\theta 2(t)}{20000}+\frac{3\,\psi 2(t)}{10}\right)+y_{\text{inb}}\,f_{\text{xB}}(t)-x_{\text{inb}}\,f_{\text{yB}}(t)-$$

$$m_{\text{zB}}(t)-\frac{3}{10}\,b_{\text{bY1}}\left(\text{y2}'(t)-\frac{1467\,\theta 2'(t)}{20000}-\frac{3\,\psi 2'(t)}{10}\right)-\frac{3}{10}\,b_{\text{bY4}}\left(\text{y2}'(t)-\frac{1467\,\theta 2'(t)}{20000}-\frac{3\,\psi 2'(t)}{10}\right)-$$

$$\frac{3}{20}\,b_{\text{bX1}}\left(\text{x2}'(t)+\frac{1467\,\phi 2'(t)}{20000}-\frac{3\,\psi 2'(t)}{20}\right)-\frac{3}{20}\,b_{\text{bX2}}\left(\text{x2}'(t)+\frac{1467\,\phi 2'(t)}{20000}-\frac{3\,\psi 2'(t)}{20}\right)-$$

$$\frac{1}{100000}\left(4979\,b_{\text{pX3}}\left(\text{x1}'(t)-\text{x2}'(t)+\frac{11\,\phi 1'(t)}{1000}-\frac{347\,\phi 2'(t)}{20000}+\frac{\psi 1'(t)}{20}-\frac{4979\,\psi 2'(t)}{100000}\right)\right)-$$

$$\frac{1}{100000}\left(4979\,b_{\text{pX4}}\left(\text{x1}'(t)-\text{x2}'(t)+\frac{11\,\phi 1'(t)}{1000}-\frac{347\,\phi 2'(t)}{20000}+\frac{\psi 1'(t)}{20}-\frac{4979\,\psi 2'(t)}{100000}\right)\right)-$$

$$\frac{1}{100000}\left(2999\,b_{\text{pY2}}\left(\text{y1}'(t)-\text{y2}'(t)-\frac{11\,\theta 1'(t)}{1000}+\frac{347\,\theta 2'(t)}{20000}+\frac{3\,\psi 1'(t)}{100}-\frac{2999\,\psi 2'(t)}{100000}\right)\right)-$$

$$\frac{1}{100000}\left(2999\,b_{\text{pY3}}\left(\text{y1}'(t)-\text{y2}'(t)-\frac{11\,\theta 1'(t)}{1000}+\frac{347\,\theta 2'(t)}{20000}+\frac{3\,\psi 1'(t)}{100}-\frac{2999\,\psi 2'(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(2999\,b_{\text{pY1}}\left(\text{y1}'(t)-\text{y2}'(t)-\frac{11\,\theta 1'(t)}{1000}+\frac{347\,\theta 2'(t)}{20000}-\frac{3\,\psi 1'(t)}{100}+\frac{2999\,\psi 2'(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(2999\,b_{\text{pY4}}\left(\text{y1}'(t)-\text{y2}'(t)-\frac{11\,\theta 1'(t)}{1000}+\frac{347\,\theta 2'(t)}{20000}-\frac{3\,\psi 1'(t)}{100}+\frac{2999\,\psi 2'(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(4979\,b_{\text{pX1}}\left(\text{x1}'(t)-\text{x2}'(t)+\frac{11\,\phi 1'(t)}{1000}-\frac{347\,\phi 2'(t)}{20000}-\frac{\psi 1'(t)}{20}+\frac{4979\,\psi 2'(t)}{100000}\right)\right)+$$

$$\frac{1}{100000}\left(4979\,b_{\text{pX2}}\left(\text{x1}'(t)-\text{x2}'(t)+\frac{11\,\phi 1'(t)}{1000}-\frac{347\,\phi 2'(t)}{20000}-\frac{\psi 1'(t)}{20}+\frac{4979\,\psi 2'(t)}{100000}\right)\right)+$$

$$\frac{3}{20}\,b_{\text{bX3}}\left(\text{x2}'(t)+\frac{1467\,\phi 2'(t)}{20000}+\frac{3\,\psi 2'(t)}{20}\right)+$$

$$\frac{3}{20}\,b_{\text{bX4}}\left(\text{x2}'(t)+\frac{1467\,\phi 2'(t)}{20000}+\frac{3\,\psi 2'(t)}{20}\right)+\frac{3}{10}\,b_{\text{bY2}}\left(\text{y2}'(t)-\frac{1467\,\theta 2'(t)}{20000}+\frac{3\,\psi 2'(t)}{10}\right)+$$

$$\frac{3}{10}\,b_{\text{bY3}}\left(\text{y2}'(t)-\frac{1467\,\theta 2'(t)}{20000}+\frac{3\,\psi 2'(t)}{10}\right)+\text{JB}_{\text{zz}}\,\psi 2''(t)==0\Big\}$$

Appendix D
Minimization Methods for Eliminating Drift.

D.1 MultiStartMin

Defining chi square with 8 variables (as sum of squares of residuals)

```
chisqr8var[a0_, a1_, a2_, a3_, a4_, a5_, a6_, myDelay_] =
```
$$\sum_{i=1}^{dataSize} ((x = d[\![i, 1]\!] ; a0 + a1\, x + a2\, x^2 + a3\, x^3 +$$

```
        a4 Exp[-a5 (x - myDelay)] Sin[a6 (x - myDelay)]
        UnitStep[x - myDelay]) - d[[i, 2]]) ^2;
```

```
res1 =
MultiStartMin[chisqr8var[a0, a1, a2, a3, a4, a5, a6, myDelay], , ,
```
$\{\{a0, 0.2\, a0_{init}, 4\, a0_{init}\}, \{a1, 4\, a1_{init}, 0.2\, a1_{init}\},$
$\{a2, 0.2\, a2_{init}, 4\, a2_{init}\}, \{a3, 4\, a3_{init}, 0.2\, a3_{init}\},$
$\{a4, 4\, a4_{init}, 0.2\, a4_{init}\}, \{a5, 0.2\, a5_{init}, 4\, a5_{init}\},$
$\{a6, 0.5\, a6_{init}, 2\, a6_{init}\}, \{myDelay, 0.5\, del_{init}, 2\, del_{init}\}\},$
```
    toler, ShowProgress → False, SimplifyOption -> False,
    CompileOption -> True, Starts → 5] // Timing
```