Motor Simulation and Parameter Identification in a Reciprocating Mechanism

by

Yun-chung Tang

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Mechanical Engineering

APPROVED:

R. G. Mitchiner, Chairman

R. G. Leonard

H. H. Robertshaw

September, 1991

Blacksburg, Virginia

Motor Simulation and Parameter Identification in a Reciprocating Mechanism

by

Yun-chung Tang

R. G. Mitchiner, Chairman

Mechanical Engineering

(ABSTRACT)

In this study, a slider-crank mechanism driven by an induction motor was studied for the purpose of deriving simulation data for the identification of the important operating parameters in the machine. First, a modification of a motor simulation program emphasizing losses is presented. The program is used to generate dynamic motor data for an analysis of a reciprocating mechanism. By analyzing the dynamic motor data, the operating parameters in the mechanism can be identified.

The purpose of the reciprocating mechanism analysis was to define an algorithm for the identification of the parameters of mass, damping, spring stiffness, and preload force of the mechanism. The time domain data (e.g. the motor net input torque, the mechanism load torque, the angular velocity, and angular acceleration, etc.) of the mechanism from a simulation can be related through the use of Newton's dynamic motion equation. By transforming the time domain data into frequency domain spectra and using a least squares algorithm, the mechanism parameters can be estimated.

The results show that the calculated mass and stiffness can be accurately solved to within 1%. While the damping and preload force may be solved within 2% and 4% respectively. The results also confirm that the separation of the load torque signal can be used in the solution technique. That is, the load torque signal of the mechanism is an arithmetic sum of the contribution of mass, damping, spring stiffness, and preload force.

The identification method of the above parameters could lead to an advancement in machine diagnosis in the future, since the operating parameters in a reciprocating machine are greatly related to an impending machine failure.

Acknowledgements

I would like to express my thanks to chairman of my advisory committee, Dr. R. G. Mitchiner, for his professional help during my graduate work and to the other committee members, Dr. R. G. Leonard and Dr. H. H. Robertshaw, for their service as committee members.

I would also like to thank my wife, Wen-jiuan Lu, and my family for their continuous support during my graduate work.

Finally, my thanks goes to the Writing Center for helping to revise this thesis.

Table of Contents

List of Illustrations

List of Tables

Chapter 1

Introduction

Reciprocating machines, such as engines and compressors, are widely used. One objective of this research is to analyze the parameters of lumped mass M (sum of slider mass M_p and the connecting rod mass of the reciprocating end), dashpot damping C, spring stiffness K, and preload force F_i, Figure 1, involved in a reciprocating mechanism powered by a three-phase induction motor. Another objective of this work is to investigate the motor loss term involved in the motor model. This model is used to develop a motor reciprocating-mechanism simulation program.

The modified motor reciprocating-mechanism simulation program is then used to create a set of useful data for the analysis of the reciprocating mechanism parameters. A study of the errors is discussed following discussion of the computation phase of the parameter identification.

In Figure 2, the motion equation of the motor reciprocating-mechanism is

$$\sum T = T_{net} - T_{load} = J_s\ddot{\theta} + C_s\dot{\theta} \tag{1.1}$$

where

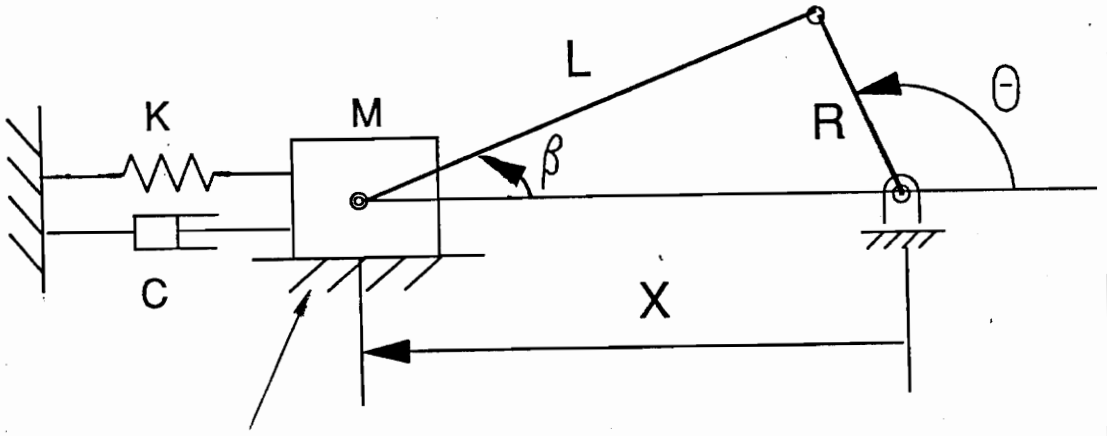


Figure 1. Reciprocating mechanism

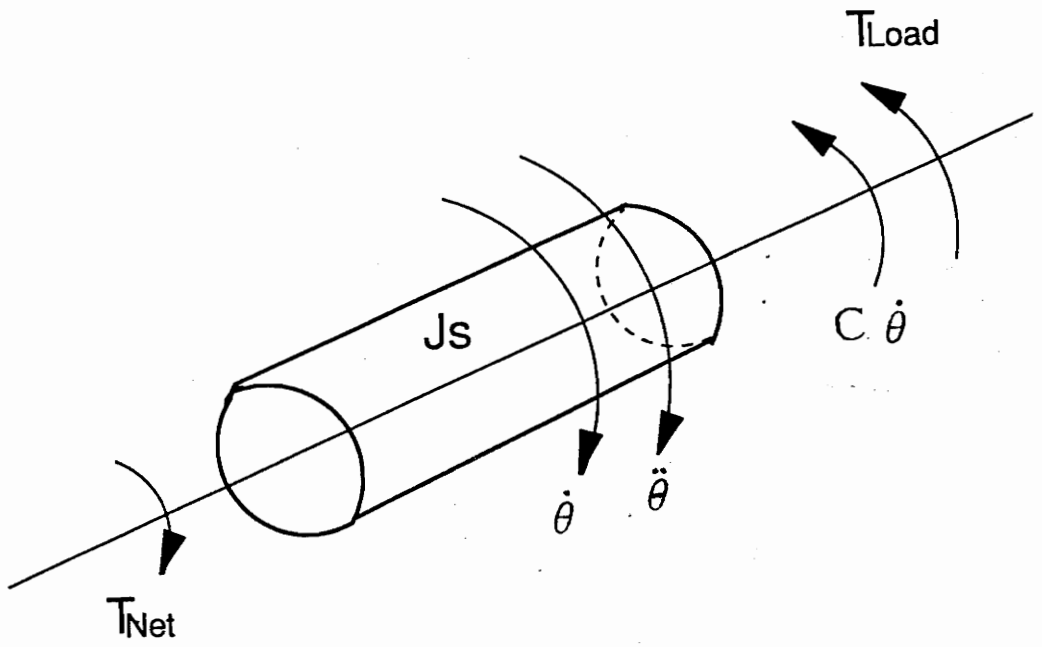


Figure 2. Free body diagram of the motor reciprocating mechanism

T_{net} = net motor output torque (N-m)

T_{load} = reciprocating mechanism load torque (N-m)

J_s = total moment of inertia of rotor, universal joint (U-joint), and crank (kg-m^2)

C_s = rotational damping (N-m-sec.)

The original version of motor simulation implements the Stanley[10] motor model and Lipo[18] equations to solve for variables such as stator and rotor phase currents, motor output electrical torque, rotor position, θ, rotor angular velocity, $\dot{\theta}$, rotor angular acceleration, $\ddot{\theta}$, stator and rotor copper losses I^2R, and instantaneous power. All of these variables are important in simulating the instantaneous behavior of the motor.

The new version of this simulation program adds the hysteresis loss and eddy current loss in the motor stator. The reason for including these losses in the simulation program is that they will profoundly influence the accuracy of the net output torque evaluation.

The modified simulation program also includes a load torque calculation subroutine for a reciprocating mechanism with parameters of the slider mass, dashpot damping, spring stiffness, and preload force. A net output torque from the motor is also developed in the code. It is stated that the reciprocating mechanism parameters M, C, K, and F_i affect the load torque calculated by the simulation program. From the load torque derivation, it is shown that the load torque is a function of these parameters, and the simulation variables $\theta, \dot{\theta}, \ddot{\theta}, X, \dot{X}, \ddot{X}$

$$T_{load} = T_{load}(M,C,K,F_i,\theta,\dot{\theta},\ddot{\theta},X,\dot{X},\ddot{X}) \tag{1.2}$$

The load torque source data used in the parameter identification is calculated according to (Equation 1.2) which is found in chapter three. The source data can be modified by changing the M, C, K, F_i in the equation.

Chapter two describes the motor operation principles, Faraday's law, Stanley's motor model and the resulting motor differential equations. A direct-and-quadrature (dq0) transformation is also discussed in this chapter. These form the basic principles and procedures for the development of the motor simulation.

Chapter three discusses the load torque of the reciprocating mechanism, the relation between linear and angular motion of the mechanism, and the dynamic analysis of the mechanism.

Chapter four discusses the losses in motor operation, focusing on the motor stator core loss. This loss, in turn, is comprised of two parts: hysteresis loss and eddy current loss. Both the losses are simulated as equivalent resistances. By considering the effects of the losses, the Stanley motor equations is modified.

Chapter five deals with the basic data processing criteria, which include the categories of the sampling process, the aliasing problem, the Nyquist frequency, and leakage prevention. This chapter also describes the motor simulation process. A detailed flow chart and description of the program are presented.

Chapter six comprises of Part One and Part Two. In Part one, an example of a harmonically excited damped system is illustrated. Part Two depicts the parameter solution technique. In particular, it covers the solution processes, from data collection, data transformation, to the final least squares algorithm solution procedure. Chapter six also compares the true and calculated results, verifying that the unknown parameters involved in the motor reciprocating mechanism can be determined by using the algorithm presented in this thesis.

Chapter seven concludes the research. Some comments are made in suggesting future work.

Chapter 2

Literature Review

2.1 Motor Introduction

The three-phase induction motor is popular because it is rugged and trouble-free compared with other types of motors. The motor is excited by a three-phase power supply. The power supply features a 120 degree phase difference between each phase of the input voltage.

A three-phase induction motor is composed of a symmetrical three-phase wound stator and a squirrel-cage shaped rotor. The stator is made up of a laminated iron core. The purpose of using iron in the stator is to provide a low-reluctance magnetic path which can set up a much more effective magnetic field for the stator winding. The reason for constructing the lamination structure is to reduce the path for eddy currents occurring in the core.

The rotor is constructed like a squirrel-cage structure filled with laminated iron plates. These plates also set up a low-reluctance path that can make a magnetic flux pass through the rotor easily.

When an applied magnetic field acts on it, an induced current occurs in the cage bars. These bars, in turn, react with the applied magnetic field without using a brush. The term "induction" comes from this reaction.

The basic concept of an induction motor is that the applied phase voltages (three-phase) set up a rotating magnetic field in the motor stator coils. Then the stator magnetic field induces a current in the motor rotor. This rotor current interacts electromagnetically with the stator magnetic field, resulting an induced torque which causes the rotor to rotate. The torque magnitude depends on the magnitude of the field, the induced current, and the relative speed between the magnetic field and the rotor itself.

2.2 *Faraday's Law*

Regarding one of the three-phase motor stator coils, at the stage of applied voltage setting up a rotating magnetic field in motor stator, the voltage has to cover the induced emf ε_a (or voltage) in stator coil a. Assuming that there is no resistance in the stator coil a, according to Faraday's law, "The induced emf in the circuit is numerically equal to the rate of change of the magnetic flux through it." [1];[1] this relation can be expressed by the equation

$$\varepsilon_a = -p\psi_a \tag{2.1}$$

where

ε_a = stator phase a induced voltage

[1] Bracketed numbers refer to references listed in the bibliography.

ψ_a = stator phase a flux linkage

p = time derivative operator, $\frac{d}{dt}$

The emf (or voltage) ε_a is induced in phase-a stator coil when a current flows in the coil. It is seen that the relation between the applied voltage and the emf in the stator coil loop is

$$v_a + \varepsilon_a = v_a - p\psi_a = 0 \tag{2.2}$$

where

v_a = stator phase a terminal voltage

In this case, the heat loss of the conductor resistance is zero because no resistance is assumed. That is, the current carrier is regarded as an ideal conductor. However, since there is always a resistance in the conductor, the above equation (Equation 2.2) has to be modified.

2.3 *Stanley[10] Motor Differential Equations*

When stator resistance is taken into account, the differential equation (phase a , for example) including conductor resistance becomes the Stanley motor equation:

$$v_a = p\psi_a + i_a R \tag{2.3}$$

where

v_a = stator phase a terminal voltage

i_a = stator phase a current

ψ = stator phase a flux linkage

R = resistance of one stator phase

p = time derivative operator $\frac{d}{dt}$

(Equation 2.3) shows that the applied voltage must provide for the flux rate change and resistance voltage drop in the motor stator. Stanley [10] assumed no core loss, i.e. the sum of hysteresis and eddy current losses, is involved in the above electromagnetic relation.

Other differential equations for the other phases can be expressed the same way. Therefore, six differential equations of basic motor voltage equations are constructed, three for the stator and three for the rotor.

According to Stanley, the flux linkage ψ is a function of the stator and rotor currents $i_{a,b,c}$ and $i_{1,2,3}$, the mutual inductance between one stator and rotor phase M, and relative angle θ between the stator axis and rotor winding. For example, flux linkage of rotor phase 1 (Figure 3) is as follows:

$$\psi_1 = li_1 + M_R(i_2 + i_3) + M(i_a \cos\theta + i_b \cos(\theta - \frac{2\pi}{3}) + i_c \cos(\theta + \frac{2\pi}{3})) \qquad (2.4)$$

where

ψ_1 = rotor phase flux linkages

l = self inductance of one rotor phase

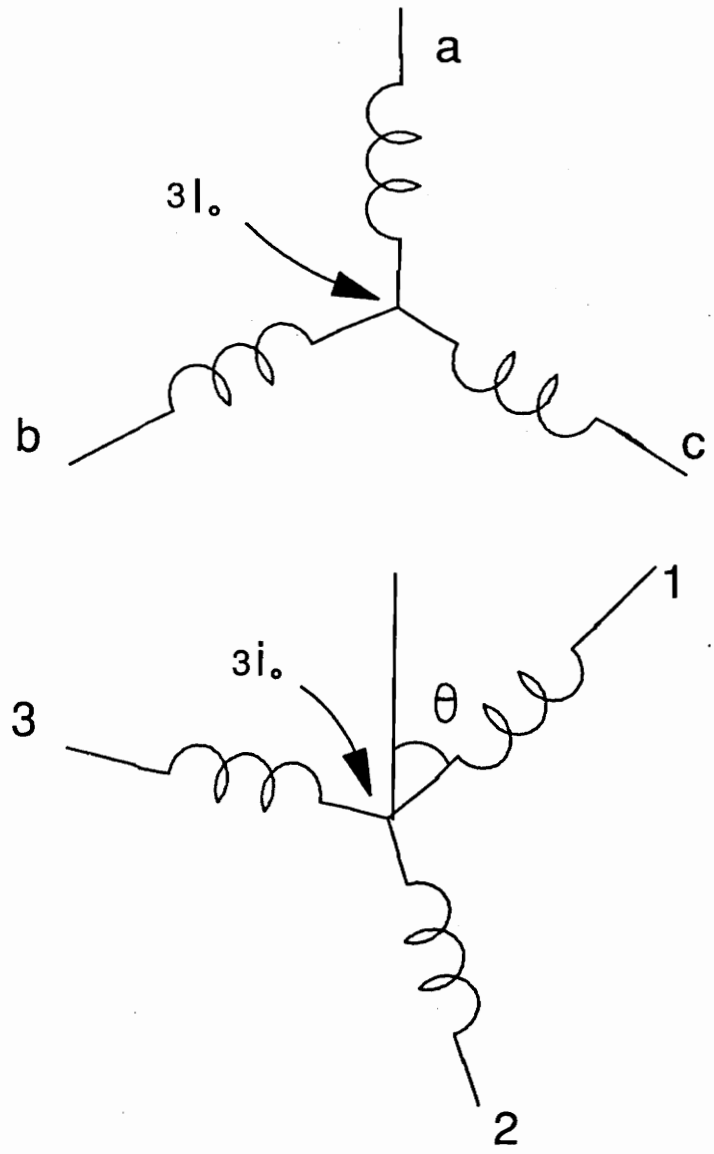


Figure 3. Relative angle between stator and rotor windings

$i_{1,2,3}$ = rotor phase currents

M = maximum value of mutual inductance between on stator phase and one rotor phase

θ = electrical angle between phase a and 1 at time t

M_R = mutual inductance between rotor phases

The flux linkage of rotor phase 1, ψ_1 (as shown in the previous page) is determined by stator currents (i_a, i_b, and i_c), rotor currents (i_1, i_2, i_3), relative angle (θ), and the inductance terms. The relative angle terms in the flux linkage equation can be eliminated by a dq0 transformation. The transformed equations can simplify the motor differential equations analysis. Through a dq0 transformation (see the following section), the stator and rotor currents can be changed to d, q directions, (Figure 4), so are the flux linkages. The transformed equations become

$$I_d = \frac{2}{3}\left(i_a - \frac{1}{2}(i_b + i_c)\right) \tag{2.5}$$

$$I_q = \frac{\sqrt{3}}{3}(i_b - i_c) \tag{2.6}$$

$$I_0 = \frac{1}{3}(i_a + i_b + i_c) \tag{2.7}$$

$$\Psi_d = L_0 I_d + M_0 i_d \tag{2.8}$$

$$\Psi_q = L_0 I_q + M_0 i_q \tag{2.9}$$

where

$L_0 = L - M_s$

L = self inductance of one stator phase

M_s = mutual inductance between stator phases

M_0 ; $M_0 = \frac{3}{2} M$, apparent three-phase mutual inductance

The above equations will lead to the expression of the electrical torque equation:

$$T_{elec} = -\frac{3}{2} I_d M_0 i_q + \frac{3}{2} I_q M_0 i_d \tag{2.10}$$

The detailed derivations are performed by Stanley[10].

Due to the existence of conductor resistance and the constancy of supplied voltage, it is obvious that the larger the resistance, the smaller the flux rate change and the induced torque.

So far the effect of core loss (the sum of hysteresis loss and eddy current loss) is not included in the differential equation. The formulas of hysteresis and eddy current losses are illustrated in the following chapter. One method of calculating the core loss with accurate results, according to Bourn[9], is called "No Load Test". As Stein[8] states, the identification of the two losses from the core loss can be achieved by using a different frequency excitation.

2.4 dq0 Transformation

According to Fitzgerald[7], a direct-and-quadrature-axis variables transformation is

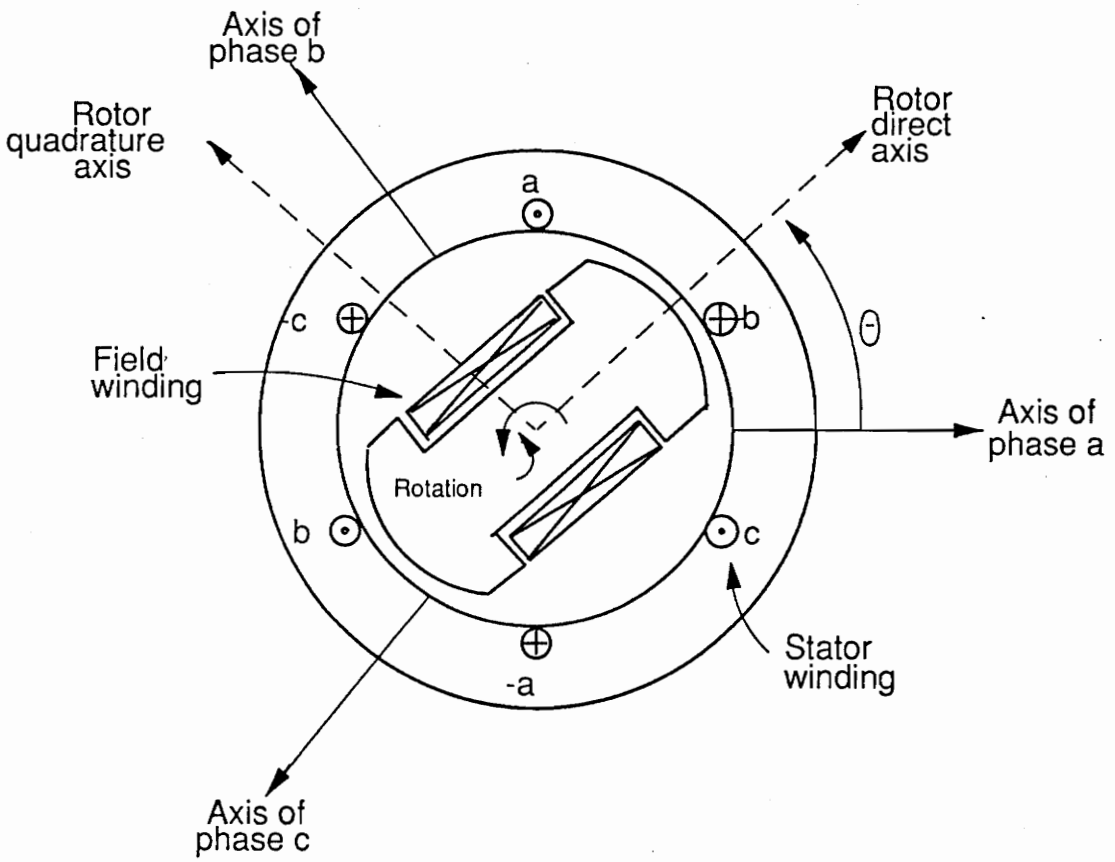


Figure 4. Relative angle between rotor direct axis and stator phase a

$$\begin{bmatrix} S_d \\ S_q \\ S_0 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos\theta & \cos(\theta - 120°) & \cos(\theta + 120°) \\ -\sin\theta & -\sin(\theta - 120°) & -\sin(\theta + 120°) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} S_a \\ S_b \\ S_c \end{bmatrix}$$

where S represents current, voltage, or flux linkage; θ represents relative angle between rotor direct axis and stator phase a (Figure 4). Its inverse transformation is as follows:

$$\begin{bmatrix} S_a \\ S_b \\ S_c \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 1 \\ \cos(\theta - 120°) & -\sin(\theta - 120°) & 1 \\ \cos(\theta + 120°) & -\sin(\theta + 120°) & 1 \end{bmatrix} \begin{bmatrix} S_d \\ S_q \\ S_0 \end{bmatrix}$$

The advantage of performing this transformation is that it simplifies the analysis from three directions (stator phases a,b,c) to two directions (d,q), since the third component $i_0 = \frac{1}{3}(i_a + i_b + i_c)$ is zero at balanced three-phase conditions. Therefore, the transformation will reduce the number of the required equations that are used to solve the motor variables.

2.5 *Parameter Identification*

2.5.1 General Parameter Identification Methods

Parameter identification is a method used to define the parameters or constants which specify a dynamic system. These properties could be mass, moment of inertia, damping constant etc. According to Åström [20], there are several ways to approach the problem.

Åström classifies the linear systems identification methods as:

1. Least squares identification: This method is a means of finding the coefficients of a fitted functional form (polynomial or nonpolynomial) when the number of available equations exceeds the number of variables to be defined (See Section 6.1.2.).

2. A probabilistic interpretation: Parameter accuracy assumptions are made in this approach. For example, the mean value of the parameter is assumed as a constant (zero, 1, k, etc.) or some kind of parameter distribution is assumed (for example, normal distribution).

3. Comparison with correlation methods: The method is a comparison of process dynamics (for example, the impulse response of a single-input single-output system) identified by the least squares method and the correlations (or cross-correlations) of the process.

4. Correlated residuals: This method is a means of checking the residual correlation of the process. It is associated with one of the following methods to solve for the parameters or constants.

5. Repeated least squares: This method is a comparison of the loss functions (refer to Equation (6.8)) from several least squares identifications of different order of the system, n (n = 1,2,...7).

6. Generalized least squares: The method is useful when both the residuals correlation and transfer function are known (Practically, these two are unknown in most cases). In this case, both the generalized error and the model parameters can be obtained from the process inputs and outputs.

7. The maximum likelihood method: This method is similar to the generalized least squares method (transfer function is known). But a likelihood function is introduced in the algorithm. It is a function of the loss function (see Equation (6.8)).

8. Instrumental variables: An instrumental matrix is used in this method. The purpose of the matrix is the simplification of the identification. The elements of the matrix are functions of

the data. With certain matrix manipulations, the elements eliminate all of the correlated residuals.

9. Levin's method: The method is only suitable for a deterministic process (That is, a mathematical model can be set up for the process) with independent observation errors. The bias due to the correlated residuals is estimated by this method.

10. Tally principle: A known random variable series is used in this method. The method implements the concept of complex conjugate to duplicate the corresponding expectations.

11. Multivariable systems: This method is more useful than the other methods to solve a system for which a suitable describing expression is not easy to find. It also includes a likelihood function in the identification algorithm.

Åström indicates that some methods could be greatly influenced by assumptions. These assumptions can be, for example, a known covariance function of the experiment errors. He also states that this influence can be eliminated by testing assumptions with experiment data.

In this research the least squares identification method (1) is employed in the parameter identification of the following:

1. Harmonically excited damped system (See Section 6.1.1).

2. Motor reciprocating-mechanism (See Section 6.2.2 & 6.2.3).

2.5.2 Least Squares Parameter Identification Method

Lim et al [21] uses the least squares algorithm to determine the dynamic parameters in a 3-joint PUMA-760 robot. The dynamic parameters to be identified are (1) mass, (2) location of center

of mass, (3) moment of inertia (of each link), (4) Coulomb and viscous friction coefficients of robot joints. Lim implements the link mass balancing concept to improve the accuracies of the estimation of moment of inertia and friction coefficient. The gravity-related terms (gravity loading on the robot joint worsens the results of parameter identification) are eliminated. Lim uses the sequential test steps[2] to reduce the number of the balanced robotic dynamic parameters. The dynamic equations are expressed by 4 kinematic parameters and 10 dynamic variables (joint torque, angular position, velocity, acceleration) of each robotic link. These dynamic equations are established using a Lagrangian formulation. The partial differential equation (2.11) of the ith link is obtained as:

$$\frac{d}{dt}\frac{\partial K}{\partial \dot{\theta}_i} - \frac{\partial K}{\partial \theta_i} + \frac{\partial P}{\partial \theta_i} + \frac{\partial D}{\partial \dot{\theta}_i} = T_i - F_{ci} \tag{2.11}$$

where

K = total kinetic energy

P = total potential energy

D = dissipation function

T_i = generalized torque

F_{ci} = Coulomb friction torque at joint i

$\dot{\theta}_i$ = angular velocity of the ith joint

---

[2] (1) The gravity-related terms are estimated first. (2) The friction terms are estimated based on the information of (1). (3) The inertia-related terms are estimated based on the information of both (1) and (2).

θ_i = angle of the ith joint

The above terms, like kinetic energy, potential energy, dissipation function etc., are expressed in the form of the measured information (like joint torques, angular positions, velocities and acceleration) and the four dynamic parameters. Then the least squares algorithm is used to identify the dynamic parameters. The results of friction coefficient, for example, show that the relative error of estimated friction coefficient ranges from 2% to 11%.

Driel et al [22] uses both the linear and nonlinear IMSL least squares routines (LSQRR, UNLSF) to identify the kinematic parameters of a 5-R-1-P (R: Revolute, P: Prismatic) robot. The parameters to be identified are the joint angular positions, position of joints, link lengths and joint offsets. Driel's observation data is produced by substituting the link parameters (link length and rotational angle) into the 5-R-1-P manipulator forward kinematic model. That is, the link parameters are input into the simulation program, then the joint values and position coordinates of the end-effector are generated. Finally, both the observation data file and the link parameters file are used in an identification program to solve for the results. Driel's simulation-based identification results show that the identification accuracy is influenced by "initial estimate of parameters, measurement accuracy and noise, encoder resolution and uncertainty, selection of measurement configurations, number of measurements, and range of notion of the joints during observations." He concludes that both IMSL routines give the same parameter estimates in the parameter identification of the 5-R-1-P robot. The only difference between the two routines is the calculation time. UNLSF consumes more time than LSQRR does (about 6 to 18 times), depending on the number of observation (The larger the number of observation, the larger the calculation time difference.)

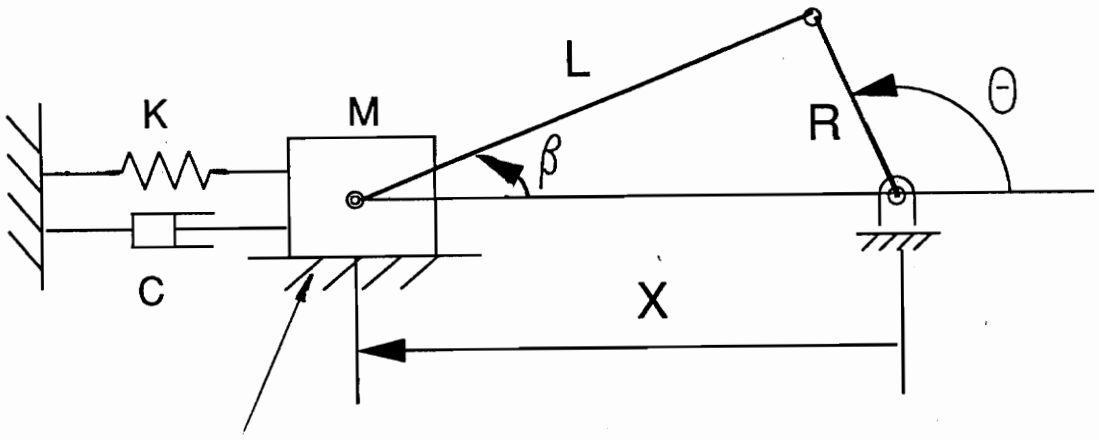
Chapter 3

Theoretical Basis

3.1 Load Torque of Reciprocating Mechanism

The reciprocating mechanism is composed of three moving parts, a crank shaft with a radius R, a connecting rod with a length L, and a piston with mass M_p. The piston is connected to the ground with a spring of a stiffness K and a dashpot of a viscous damping constant C. A friction coefficient μ exists between the piston and the slideway.

The first part of this chapter discusses the motion relationships between the piston and crank shaft. This relation, used to calculate load torque, is presented in the second part of the chapter.



dynamic friction coefficient = μ

Figure 5. Reciprocating mechanism

3.2 *Relation Between Linear Motion of Piston and Angular Motion of Crank*

This section establishes the relationship between the linear motion of piston $X, \dot{X}, \ddot{X}$, and angular motion of crank $\theta, \dot{\theta}, \ddot{\theta}$, respectively. The objective of this process is to define the piston variables, $X, \dot{X}, \ddot{X}$, in terms of crank variables, $\theta, \dot{\theta}, \ddot{\theta}$, respectively, since the piston variables can not be measured as easily as the crank variables.

3.2.1 Equation of piston displacement

With reference to Figure 5, the piston displacement X can be expressed by

$$X = -R\cos\theta + L\cos\beta \tag{3.1}$$

where

R = crank shaft arm radius

L = connecting rod length

θ = instantaneous angle between crank shaft arm and positive x axis direction.

β = angle between connecting rod and positive x direction.

According to sine law, the relation of L, R, β, θ is

$$\frac{\sin\beta}{R} = \frac{\sin(\pi - \theta)}{L} \tag{3.2}$$

Therefore,

$$\sin\beta = \frac{R\sin\theta}{L} \tag{3.3}$$

The displacement of the piston mass relative to the crank rotating center is

$$X = -R\cos\theta + \sqrt{L^2 - R^2\sin^2\theta} \tag{3.4}$$

When the mechanism is at a stable static equilibrium state, $\theta = 0$, the spring compressed deformation is δ (the preload force comes from this spring compressed deformation δ), as shown in Figure 6. Therefore, the preload force F_i of the piston is $K\delta$; thus the equilibrium displacement of piston

$$X_{equilibrium} = L - R \tag{3.5}$$

ΔX is defined as a displacement difference from the equilibrium position:

$$\Delta X = X - X_{equilibrium} \tag{3.6}$$

Substituting the piston displacement equation (Equation 3.4) into the above equation (Equation 3.6), the displacement difference equation becomes

$$\Delta X = -R\cos\theta + \sqrt{L^2 - R^2\sin^2\theta} - (L - R) \tag{3.7}$$

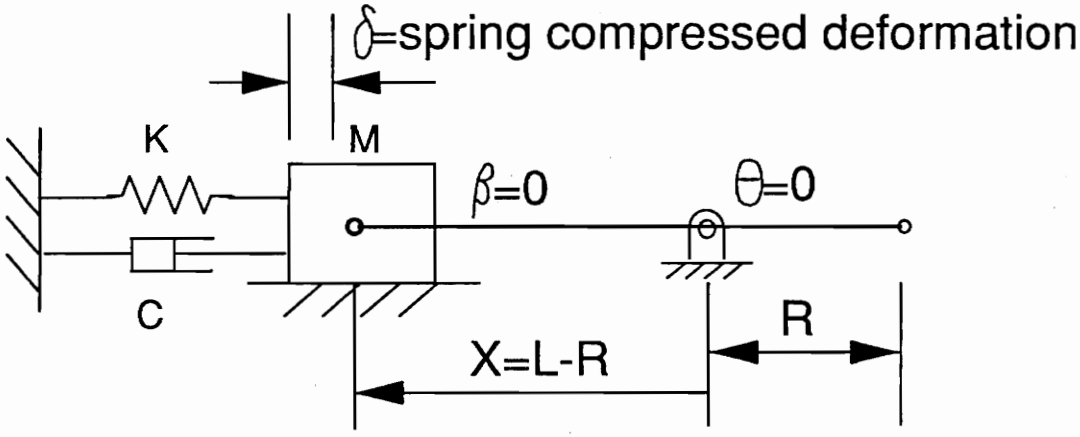


Figure 6. Static equilibrium position of reciprocating mechanism

3.2.2 Equation of Piston Velocity and Acceleration

The cosine law gives

$$L^2 = R^2 + X^2 + 2RX\cos\theta \tag{3.8}$$

Differentiating both sides, we obtain

$$\frac{d}{dt}L^2 = \frac{d}{dt}(R^2 + X^2 + 2RX\cos\theta) \tag{3.9}$$

resulting in

$$X\dot{X} = R(-\dot{X}\cos\theta + X\dot{\theta}\sin\theta) \tag{3.10}$$

Differentiating again, we obtain

$$\dot{X}^2 + X\ddot{X} = R(-\ddot{X}\cos\theta + X\dot{\theta}^2\cos\theta + X\ddot{\theta}\sin\theta + 2\dot{X}\dot{\theta}\sin\theta) \tag{3.11}$$

Rearranging the above velocity and acceleration equations (Equations 3.10, 3.11), the piston displacement, velocity, and acceleration become

$$X = -R\cos\theta + \sqrt{L^2 - R^2\sin^2\theta} \tag{3.12}$$

$$\dot{X} = \frac{X\dot{\theta}R\sin\theta}{R\cos\theta + X} \tag{3.13}$$

and

$$\ddot{X} = \frac{-\dot{X}^2 + X\dot{\theta}^2R\cos\theta + X\ddot{\theta}R\sin\theta + 2\dot{X}\dot{\theta}R\sin\theta}{R\cos\theta + X} \tag{3.14}$$

The above equations are in the form of

$$F_X(\theta, X) = 0 \tag{3.15}$$

$$F_{\dot{X}}(\theta, \dot{\theta}, X, \dot{X}) = 0 \tag{3.16}$$

$$F_{\ddot{X}}(\theta, \dot{\theta}, \ddot{\theta}, X, \dot{X}, \ddot{X}) = 0 \tag{3.17}$$

where $\theta, \dot{\theta}, \ddot{\theta}$ are considered as easily measured parameters. Refer to Kanth[17] and Williams[14].

3.2.3 Dynamic Analysis

The dynamic analysis can be simplified by assuming that the mass of the connecting rod is distributed at the rod ends, the crank shaft and the piston. A part of the connecting rod is added to the piston mass to form a lumped mass M. Therefore, the connecting rod is regarded as massless. (Williams[14])

Through the dynamic analysis of the mechanism, as shown in the following figures, the free body diagrams of the mechanism components reveal the relation between the load torque and piston displacement, velocity, and acceleration.

(1) Crankshaft (Figure 7) : the torque equation is as follows:

$$R(F_{Rx} \sin\theta - F_{Ry} \cos\theta) - T_{load} = 0 \tag{3.18}$$

(2) Connecting rod (Figure 8) : the torque equation and force equations are as follows:

$$F_{Rx} - F_{Lx} = 0 \tag{3.19}$$

$$F_{Ly} - F_{Ry} = 0 \tag{3.20}$$

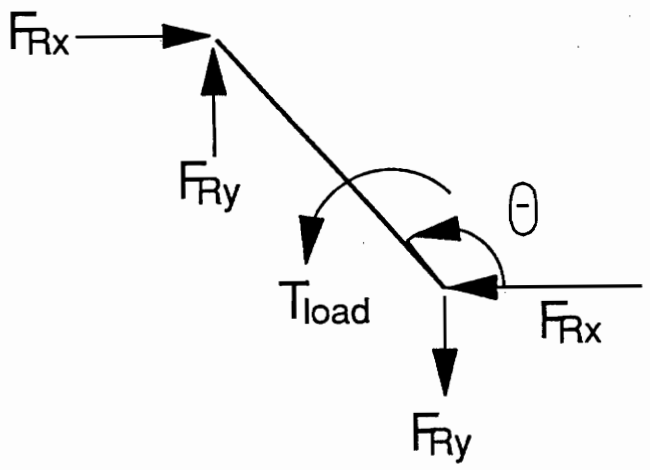


Figure 7. Free body diagram of mechanism crank

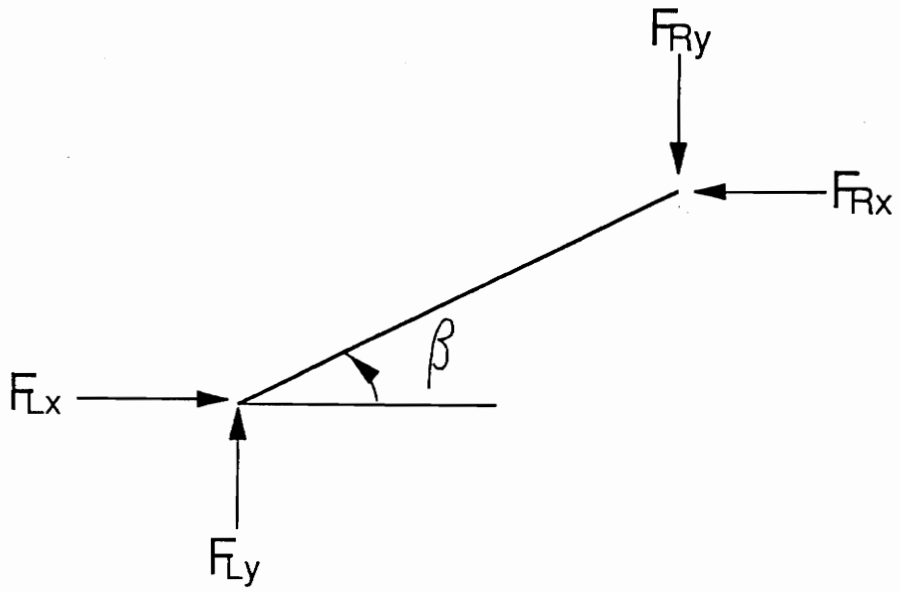


Figure 8. Free body diagram of mechanism connecting rod

$$L(F_{Ly}\cos\beta - F_{Lx}\sin\beta) = 0 \tag{3.21}$$

(3) Piston (refer to Figure 9) : the horizontal forces acting on the piston are

a. Connecting rod force F_{Lx}

b. Damping force $C\dot{X}$

c. Spring force $K\Delta X$

d. Preload force F_i $(= K\delta)$

and

e. Friction force $\mu(F_{Ly} + Mg)$.

Applying Newton's second law of motion to the mass M yields

$$M\ddot{X} = \sum F = F_{Lx} - \mu(F_{Ly} + Mg) - C\dot{X} - K\Delta X - F_i \tag{3.22}$$

Solving the above equations from (3.18) to (3.22) yields the load torque equation

$$T_L = \frac{R}{1 - \mu\tan\beta}[(\sin\theta - \cos\theta\tan\beta)(M\ddot{X} + C\dot{X} + K\Delta X + F_i + \mu Mg)] \tag{3.23}$$

where

M = sum of the lumped mass of connecting rod and piston mass (kg)

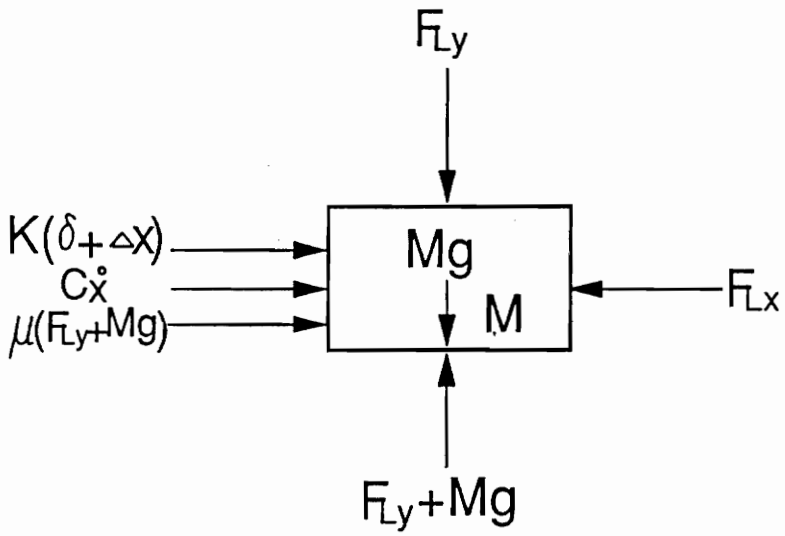


Figure 9. Free body diagram of mechanism piston

C = dashpot damping (N-sec./m)

K = spring stiffness (N/m)

μ = the friction force coefficient between piston and slideway

g = acceleration of gravity (9.81 m/ $sec.^2$)

Tan β can be expressed as

$$\tan\beta = \frac{\sin\theta}{\sqrt{\left(\frac{L}{R}\right)^2 - \sin^2\theta}} \tag{3.24}$$

Since the load torque T_L is a function of θ, X, $\dot{X}$, $\ddot{X}$, the F_X, $F_{\dot{X}}$, $F_{\ddot{X}}$ functions can be used to determine these parameters by substituting θ, $\dot{\theta}$,$\ddot{\theta}$, respectively, into these equations.

Chapter 4

Motor Losses

In this simulation, a one-half hp, 3-phase induction motor is used. Due to its small output power level, the loss fraction is not negligible.

The losses in induction motors occur as (a) stator copper loss, or heating loss (phase *a*, for example):

$$P_{s,c} = i_a^2 Rs \tag{4.1}$$

and (b) rotor copper loss (phase 1 for example),

$$P_{r,c} = i_1^2 Rr \tag{4.2}$$

(c) stator core loss, and as (d) mechanical loss. Because the Stanley simulation program already quantified the stator and rotor copper losses, the following sections are dedicated to the identification of the stator core loss. The stator core loss is comprised of two parts, the hysteresis loss and the eddy current loss.

4.1 *Hysteresis Loss*

The hysteresis loss is caused by the properties of the ferromagnetic material used in the stator core. Although the material reduces the reluctance of the space in the stator windings, its magnetization and demagnetization processes result in an irreversible energy loss. We assume that the equivalent resistance of this loss is termed R_h .

According to Chapman[4], the hysteresis loss of magnetic materials is shown as

$$P_h = K_h f (B_{\max})^x \tag{4.3}$$

where

P_h = hysteresis loss rate per unit volume of stator core, in W/m^3

K_h = constant for a given material

x = approximately equals 1.6 for most usual materials, at flux densities of 0.15 to 1.4 Weber/ m^2

$B_{\max}$ = maximum value of flux density (Weber/m^2)

f = frequency of variation of flux, Hz

The work W_m, done by the magnetic intensity H , from point i to point j, is

$$W_m = \int_{B_i}^{B_j} H dB \tag{4.4}$$

where B is the flux density, H is the magnetic intensity of the stator core directly related to the stator phase current i_a.

The hysteresis loop shown in Figure 10 can be divided into two parts, energy storing and energy releasing.

(1) Energy storing:

point 1 to point 2 : where H and dB are both positive;

point 3 to point 4 : where H and dB are both negative.

(2) Energy releasing:

point 2 to point 3 : where H is positive, dB is negative;

point 4 to point 1 : where H is negative, dB is positive.

The net amount of energy of path 1-2-3 and path 3-4-1 are the same due to the symmetry of the hysteresis loop with respect to the origin.

For a 60 Hz alternating field, the above hysteresis loop also magnetizes and demagnetizes at 60 times per second. It is obvious that in this case, the hysteresis loss of the stator core is 60 times the loop area per second. For the loop shape of Figure 10, the energy releasing part can be neglected.

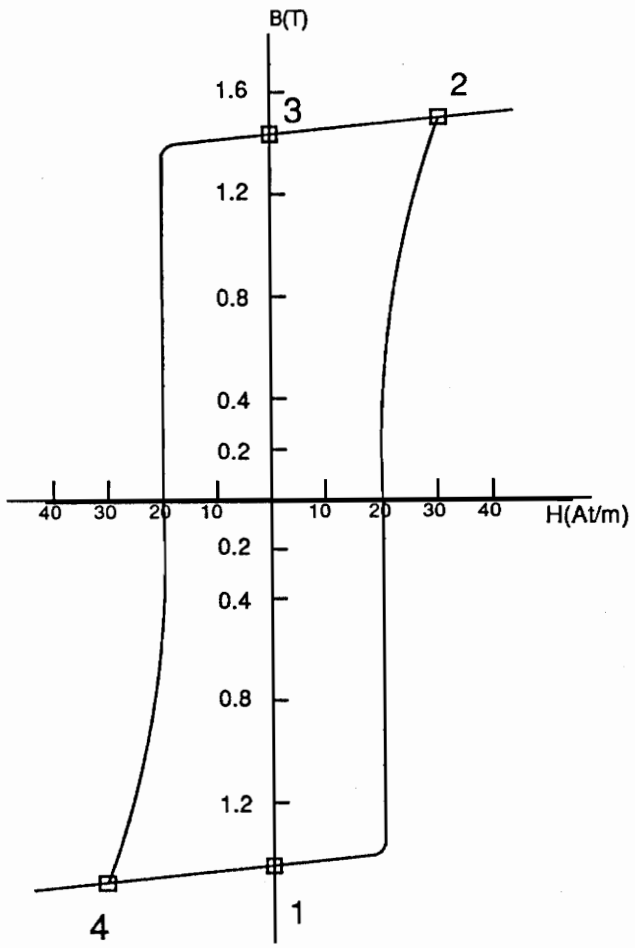


Figure 10. Hysteresis loop

The energy storing process takes place every other half cycle and lasts for one fourth of one sixtieth of a second. The average power of the loss can be assumed as

$$P_h = i_a^2 \times R_h \tag{4.5}$$

Although the energy is temporarily stored during 1-2, 3-4, it eventually turns into heat. The energy storing process occurring in the stator is quite different from the one taking place in the inductor. Most of the energy stored in the core changes to heat, a small portion of the stored energy turns back to electrical energy, depending on the material used. Therefore, the hysteresis loss works like a resistor in the stator circuit.

4.2 *Eddy Current Loss*

Eddy current loss is caused by flux change in the stator core, since a changing flux induces an emf in the iron core. Unlike the hysteresis loss, the eddy current loss keeps on dissipating energy; therefore, it is treated as a resistance current in the stator circuit. Assuming the equivalent resistance of this loss is termed R_e, eddy current loss can be expressed as power per unit volume of core, according to Cochran[2].

$$P_e = \frac{\pi^2 f^2 B_{\max}^2 t^2}{6\rho} \tag{4.6}$$

where

P_e = eddy current loss rate per unit volume of stator core, in W/m^3

ρ = resistivity of the material

t = lamination thickness

4.3 *Quantification of the Losses*

Since most of the experimental results show that the sum of the above losses, i.e., the core loss, instead of the separate hysteresis and eddy current losses, the losses may be distinguished by a variation of frequency.

By changing the exciting source frequency and keeping the maximum flux density constant, the hysteresis loss and eddy current loss can be identified empirically, because P_h is proportional to f and P_e is proportional to f^2.

After the average amount of hysteresis and eddy current losses are checked, we can continue the dynamic analysis. Due to the nature of hysteresis and eddy current losses consuming energy in the dynamic electrical circuit, their effects are to be analyzed.

Because they consume energy like a resistor, the equivalent resistances for the hysteresis and eddy current losses are assumed to be R_h and R_e respectively.

4.3.1 Quantification of Hysteresis Equivalent Resistance

Since Bourn[9] experimentally found that the average core loss was 6 percent of the input power for the motor used in the simulation, thus the average hysteresis loss can be found by the difference of the core loss and the eddy current loss:

$$P_h = P_{core} - P_e \tag{4.7}$$

where

$$P_e = \frac{\pi^2 f^2 B_{\max}^2 t^2}{6\rho} \tag{4.8}$$

and

$$B_{\max} = \frac{V_{rms}}{4.44NfA} \tag{4.9}$$

where

V_{rms} = root mean square voltage

N = coil turns

f = voltage frequency

A = cross section area of magnetic core

The hysteresis loss P_h can be expressed in the form of a resistance loss equation

$$P_h = i_a^2 R_h \tag{4.10}$$

where R_h is the average equivalent resistance of the loss. As mentioned before, if the hysteresis loop is of the shape shown in Figure 10, there is almost no energy released during paths 2-3 and 4-1. That is, the inductance effect in the core may be neglected. So the loss should be regarded as a resistance load instead of an inductance-involved load.

Since the heat loss of the hysteresis loop mainly happens along paths 1-2 and 3-4, like a resistor, it is reasonable to assume R_h is a constant when it is considered "active" along these paths.

Before quantifying the time-variant magnitude of $R_h(t)$, it is necessary to find out the effect from the phase current i_a which directly relates to the variation of magnetic intensity H.

4.3.2 Phase Shift and Power Factor

A 0.46 power factor (P.F., see Sears et al [1]), the same as in the real machine, is used in the simulation program. A non-unit power factor means that there is a phase shift θ between the voltage and the current:

$$\theta = \cos^{-1}(0.46) = 62.6deg. = 1.093rad.$$

Since the loop is symmetrical with respect to the origin, the effect of the path 1-2-3 is the same as that of the path 3-4-1. Furthermore, $R_h(t)$ along path 1-2 is considered "active" (i.e. heat loss happens) , and 2-3 is "dormant" (i.e. no heat loss), so $R_h(t)$ is $2R_h$ along 1-2 and zero along 2-3.

So the time-variant hysteresis resistance $R_h(t)$ can be expressed as

$$R_h(t) = 2R_hF(t) \tag{4.11}$$

where R_h is the average equivalent hysteresis resistance.

The value of $F(t)$ is either one or zero, depending on the following formula:

$$g(t) = INTEGER\{(\frac{1}{FREQ} - \frac{\cos^{-1}(P.F.)}{2\pi(FREQ)} + t)/(\frac{1}{FREQ})\} \tag{4.12}$$

where

$FREQ$ = phase current frequency

If $g(t)$ is odd, then $F(t)$ is one; otherwise, $F(t)$ is zero.

4.4 *Motor Differential Equations with Loss Term*

The modified Stanley motor equation (Equation 4.13) of stator phase a, including core loss, i.e., hysteresis plus eddy-current losses, can be changed to

$$e_a = p\psi_a + (R + R_h + R_e)i_a \tag{4.13}$$

where R_h and R_e are the simulated resistances to hysteresis and eddy current "loads". It is obvious that the flux rate $p\Psi_a$ would be smaller than that of the case which neglects both losses. The difference between the original D.E. and the modified one is the resistance voltage term. By the manipulation mentioned in the discussion of the original D.E., the modified equation becomes

$$p\Psi_d = E_d - (R + R_h + R_e)I_d \tag{4.14}$$

The equivalent resistances of the hysteresis and the eddy current losses R_h, R_e should be determined experimentally since the amount of loss depends on the material used in the stator core.

Chapter 5

Data Acquisition and Simulation

5.1 Sampling

In the simulation stage, time-history signals are produced by a simulation program which generates a set of data that can be used to solve for the system parameters. This is a simulation of the actual machine where these signals are produced and acquired. The data produced by the code is discrete rather than continuous. Some basic considerations about this sampling process are discussed. Assuming the data record of period T, then the frequency resolution (i.e. minimum frequency interval required to distinguish two different frequency spectra) is $\Delta f = \frac{1}{T}$. At least two points for the definition of a cycle are required to sample a periodic signal. That is, the minimum sampling frequency $f_{\min}$ is two times the signal frequency. Therefore, the minimum number of data points required to describe the data record is

$$N = f_{\min} T = \frac{f_{\min}}{\Delta f}$$

where

N = minimum number of data points

It follows that the maximum sampling time interval for equally spaced sample record is $\Delta t = \frac{1}{f_{\min}}$. For example, suppose that a time history record of period 0.4 sec. and its Fourier transform exists only within a range 0 to 125 Hz, and is zero at all other frequencies. Then the minimum number of the sample record is $2BT = 2(125)(0.4) = 100$ samples (where B, the bandwidth, in this case, B = 125 Hz, minimum sampling rate = 250 Hz.) The time interval Δt = 1 /(2(125)) is 0.004 sec. If the sampling rate is less than the minimum sampling rate 2BT, aliasing occurs; thereby changing a high frequency component to a low frequency one. Therefore, the highest frequency that can be detected by a sampling rate $f_s = \frac{1}{\Delta t}$ is $f_{nyq} = \frac{1}{2\Delta t}$ Hz. This highest frequency f_{nyq} is called the Nyquist frequency, or folding frequency.

5.2 *Aliasing*

If the sampling rate is less than the minimum sampling rate, $f_{\min}$, then aliasing occurs. Aliasing changes a high frequency component to a low frequency one. Aliasing will contaminate the true frequency spectra. Whenever an aliasing happens, either a higher sampling rate (half of the sampling rate is at least the largest aliased frequency signal) or a low pass filter (it removes high frequency signals) will be needed to improve the situation; (Bendat[11].)

The folded-back original frequency signal will contaminate the true spectrum, producing the so-called aliasing error, as discussed in the previous section. For Nyquist frequency $f_{nyq} = \frac{1}{2\Delta t}$, the

spectrum of any frequency f (where $0 \leq f \leq f_{nyq}$) is mixed with the higher frequencies. Those higher frequencies which aliase with f are defined by $2nf_{nyq} \pm f$ (where n = any integer)

To cure the aliasing error, either the sampling rate can be increased or a low pass filter (so called anti-aliasing filter) can be applied during sampling.

5.3 *Leakage*

Leakage is a symptom of distortion in the spectral representation of the data, due to the existence of non-periodic (or truncated) functions in the sampling time domain window. Because the Fourier transform of a non-periodic waveform is a continuous function of frequency, the waveform is equivalent to a summation of sinusoids of all frequencies. Therefore, it is quite possible that the Fourier transform of a periodic waveform looks like a transform of a non-periodic waveform, due to a non-multiple-period sampling or a discontinuous sampled waveform; (Brigham[12].)

This leakage problem can be cured by sampling an integral number of periods. Since a reciprocating machine operates periodically, we can take advantage of this property.

5.4 *Descriptions of Programs*

5.4.1 Motor and Reciprocating Mechanism Simulation Program

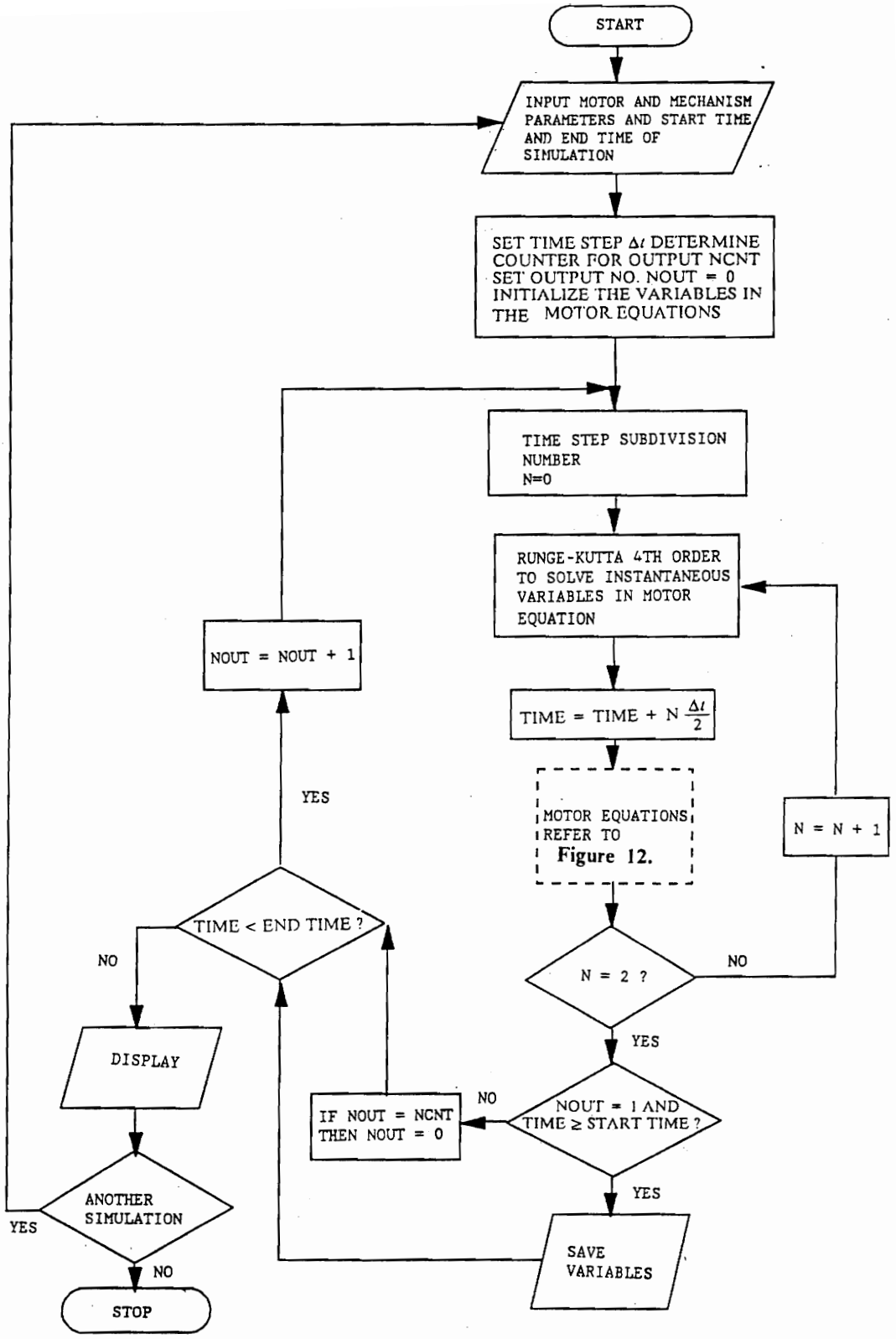


Figure 11. Motor simulation program flow chart

As shown in the motor simulation program flow chart in Figure 11, the motor parameters, the reciprocating mechanism parameters, and the simulation time are fed into the program first. The motor parameters are included with stator coil and rotor coil resistances (R1,R2), linkage inductances (X1,X2), magnetizing inductance (XM), total rotary moment of inertia (J), voltage frequency (FREQ), and number of poles (NPOLES).

The reciprocating mechanism parameters are comprised of the lumped mass of the piston and the reciprocating end of the connecting rod (M), the dashpot damping (C), the spring stiffness (K), and the preload force (F).

Then, the time step (Δt), counter number for output (NCNT), output number (NOUT), and initial values of the variables of motor differential equations are determined. The time step defined in the program is 1/9600 second. This small time step is used to calculate accurate variables in the simulation program. NCNT is used to determine the number of the time step for saving the variables. Each time when NOUT equals 1, the variables are saved. When NOUT equals NCNT, NOUT is reset to zero.

A fourth-order Runge-Kutta-Gill integration subroutine is used to solve motor differential equations; (Cheney[15].) Each time step is divided into two sub-steps to find the value of the differential terms. After each execution, the variables are used in the motor differential equations to calculate new variables.

After executing one time step, if NOUT equals 1, the variables will be saved. If the time is less than the end time, the simulation will continue.

The motor differential equations subroutine is shown in Figure 12. As soon as the variables are determined, the instantaneous load torque and the hysteresis loss ("dormant" or "active") are cal-

culated. The above values are substituted into the motor differential equations for the next Runge-Kutta subroutine calculation. The stator and rotor phase currents and phase voltages are also calculated in this subroutine.

The simulation times are the starting time and the ending time of the simulation. The starting time is chosen to be after the motor is operating at steady-state. The ending time, in turn, is chosen according to the sampling rate. As discussed in the previous section, we want to take advantage of the deterministic characteristics (i.e. the signals can be expressed mathematically, since they are periodic functions) of the motor signals, no matter whether it is load torque, motor net output torque, angular velocity or whatever.

Since the time step defined in the program is fixed, the time interval, ending time minus starting time, determines the sampling rate. Since the number of data points stored is within one thousand, the larger the time interval, the lower the sampling rate. Although not used in the sampling process, the maximum sampling rate which can be used in this program is the reciprocal of the fixed time step. Although this maximum sampling rate can pick up the most detailed and comprehensive information, it may not greatly improve the results. The frequency spectra of the transformed time domain signals show that there are some spectra, usually in the low frequencies, that dominate the motor-reciprocating-mechanism operation.

The sampling rate used in generating data is 1920 Hz. This frequency is obtained by setting the time interval at 0.5 second. There are about 7 (this number slightly changes when the load torque differs) revolutions occurring during the 0.5 sec. time interval.

Some of the simulated time-domain results are shown from Figure 13 to Figure 25.

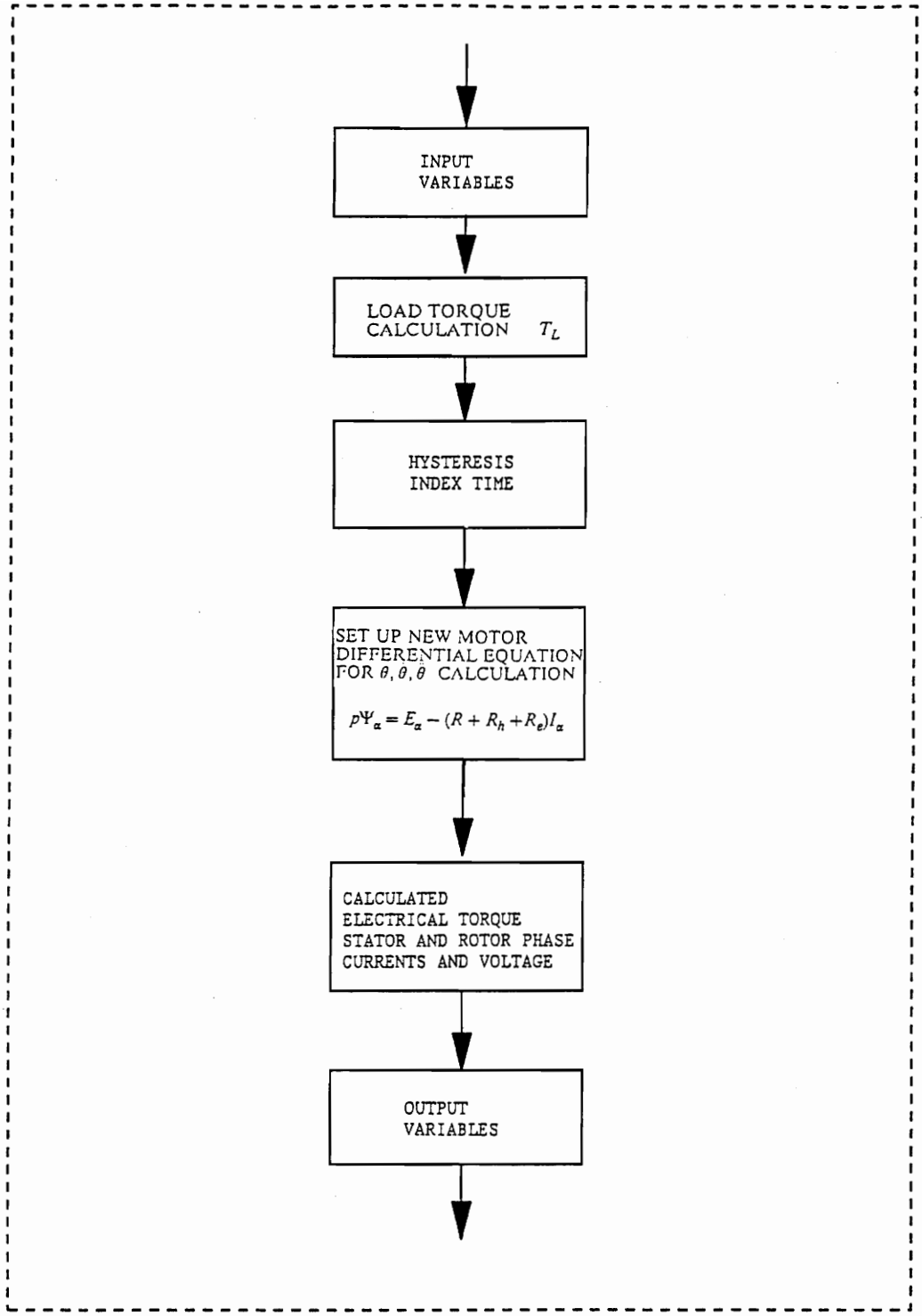


Figure 12. Motor differential equation flow chart

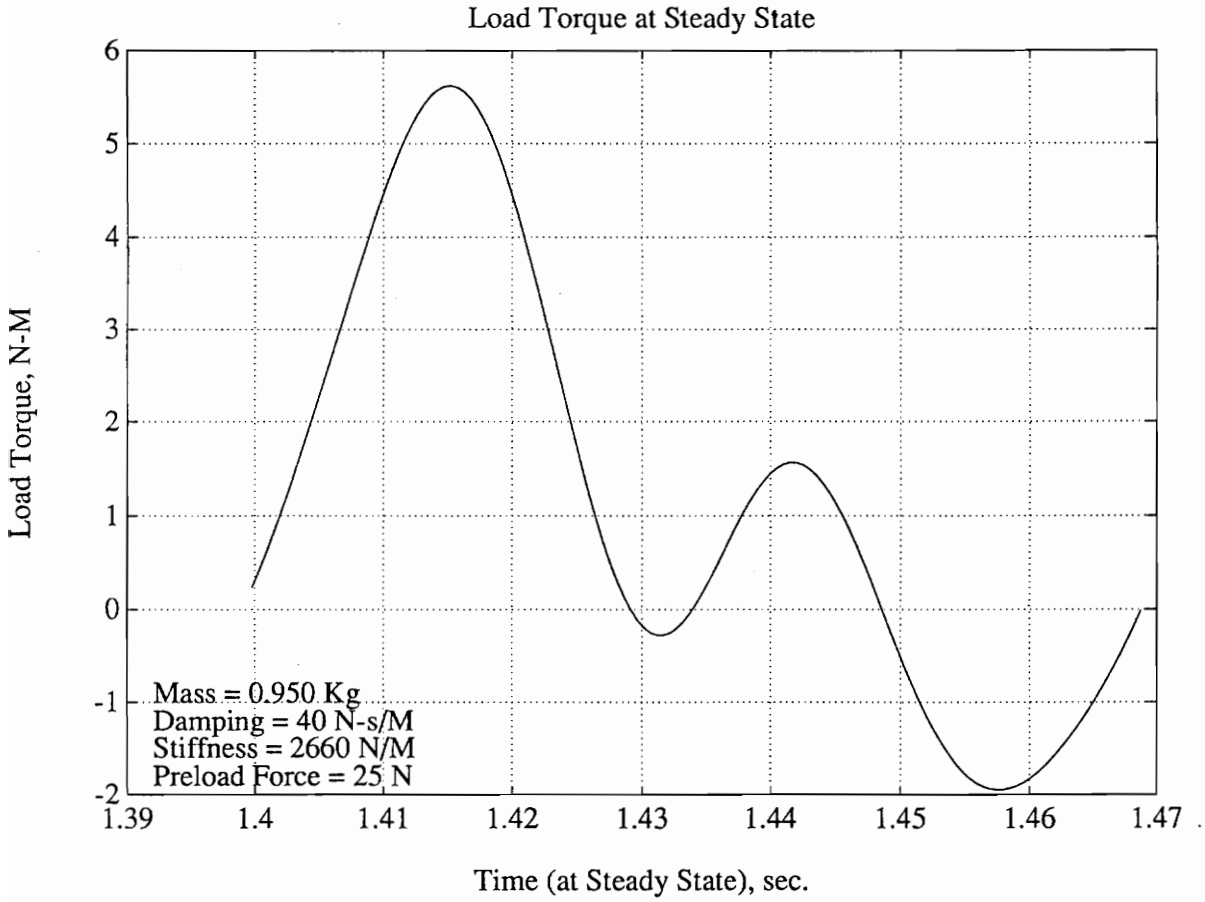


Figure 13. **Load torque of motor-reciprocating mechanism at steady state**

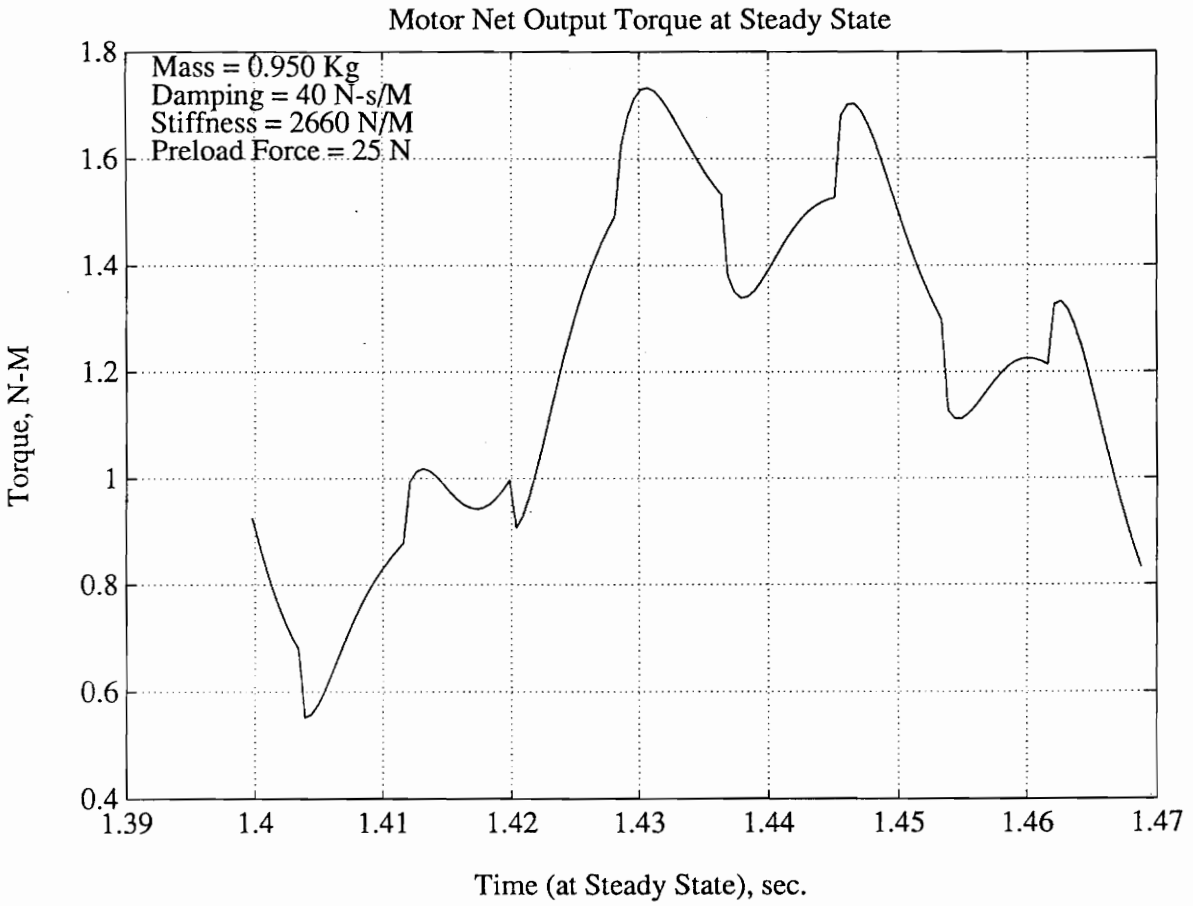


Figure 14. Net output torque of motor-reciprocating mechanism at steady state

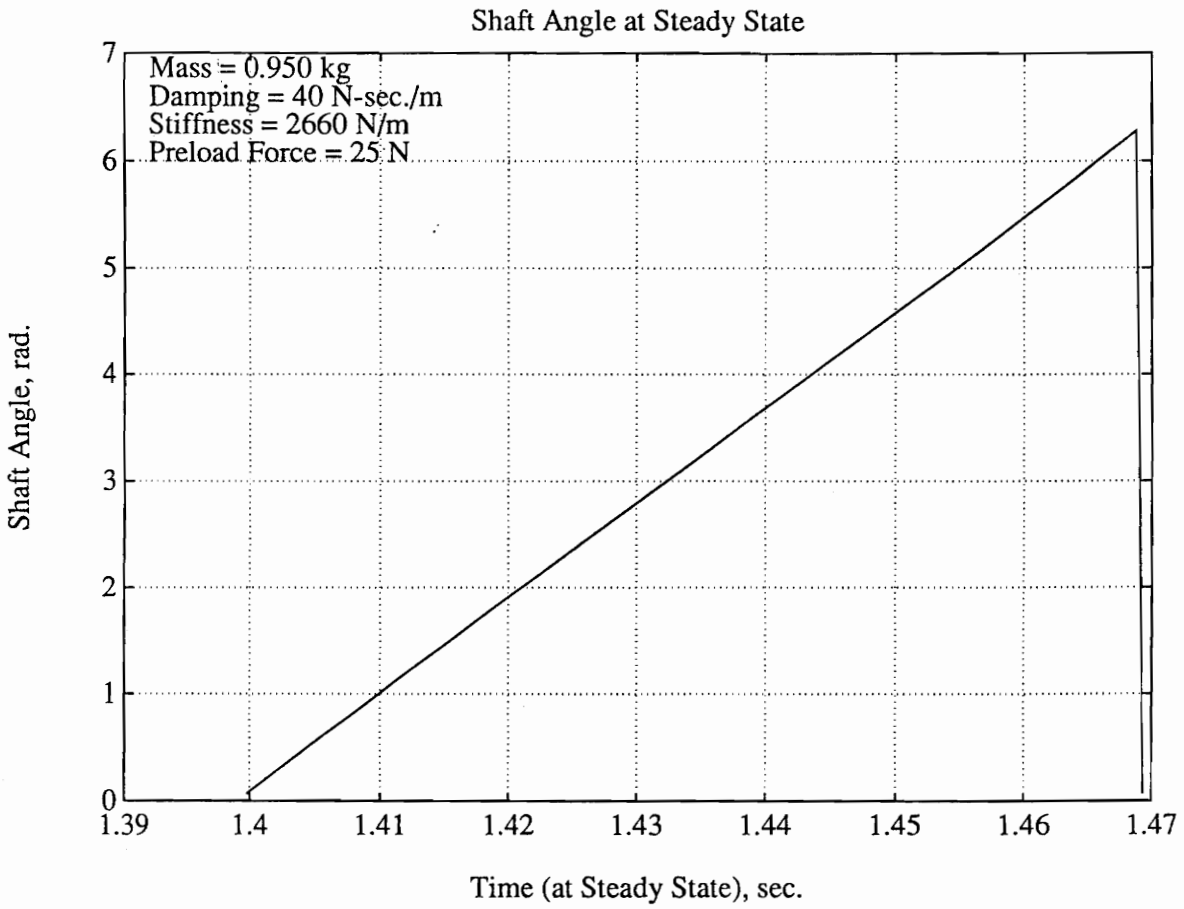


Figure 15. Shaft angle of motor-reciprocating mechanism at steady state

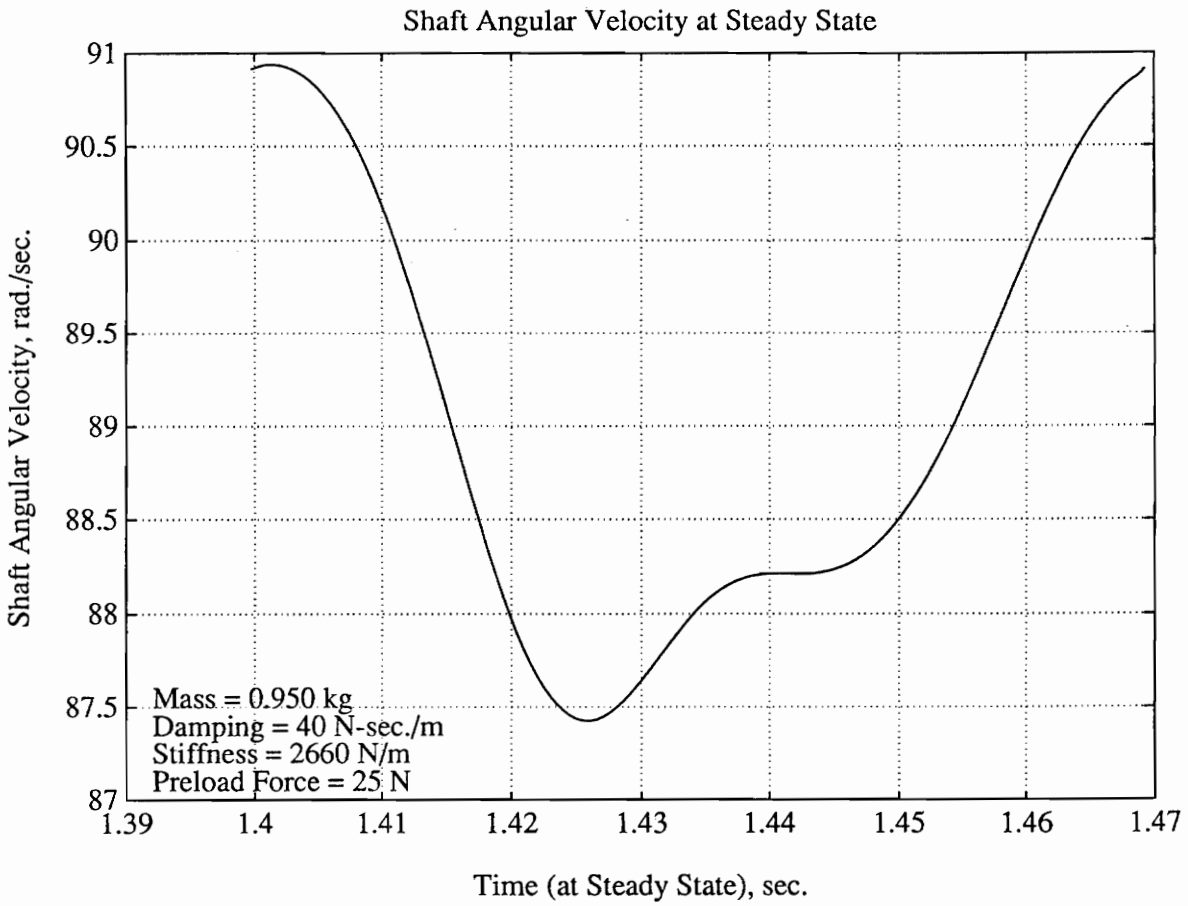


Figure 16. Angular velocity of motor-reciprocating mechanism at steady state

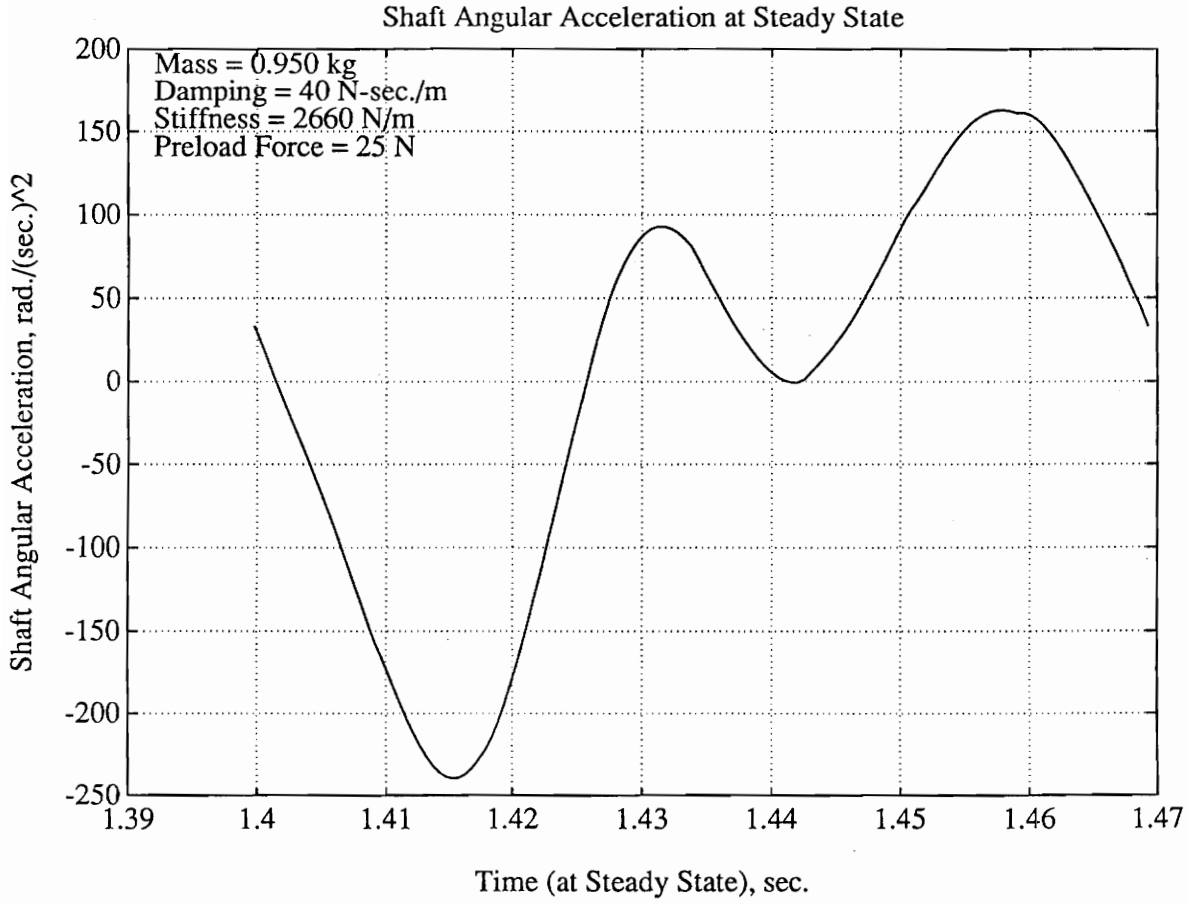


Figure 17. Angular acceleration of motor-reciprocating mechanism at steady state

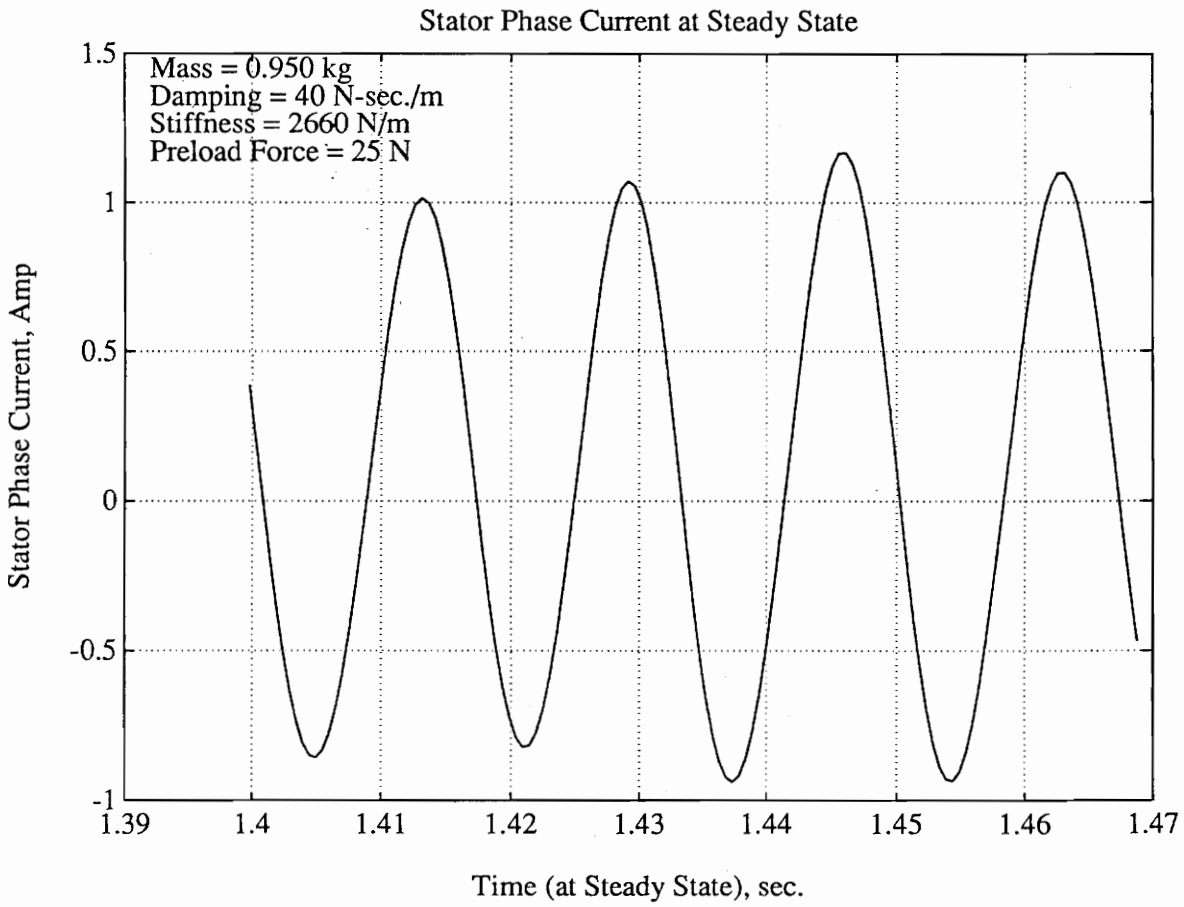


Figure 18. Phase current of motor-reciprocating mechanism at steady state

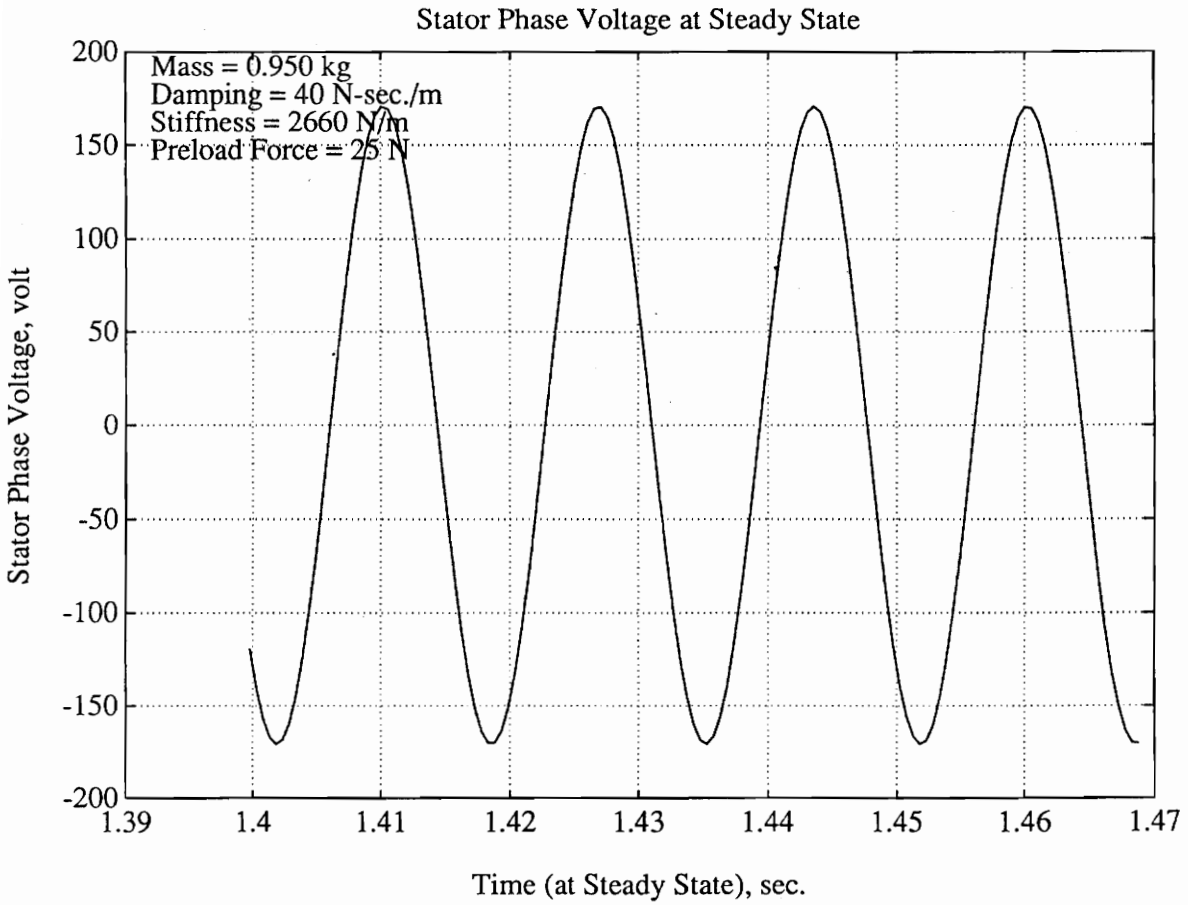


Figure 19. Phase voltage of motor-reciprocating mechanism at steady state

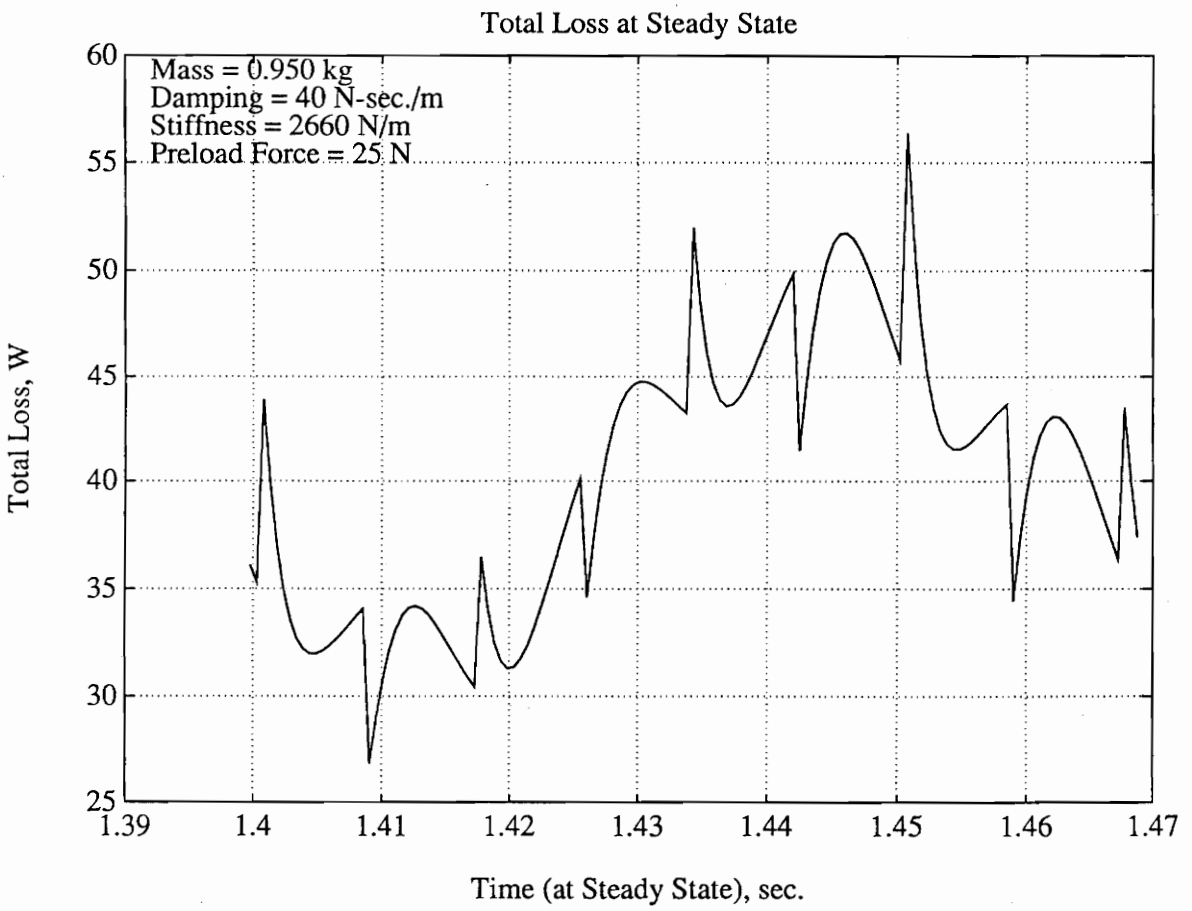


Figure 20. Total losses of motor-reciprocating mechanism at steady state

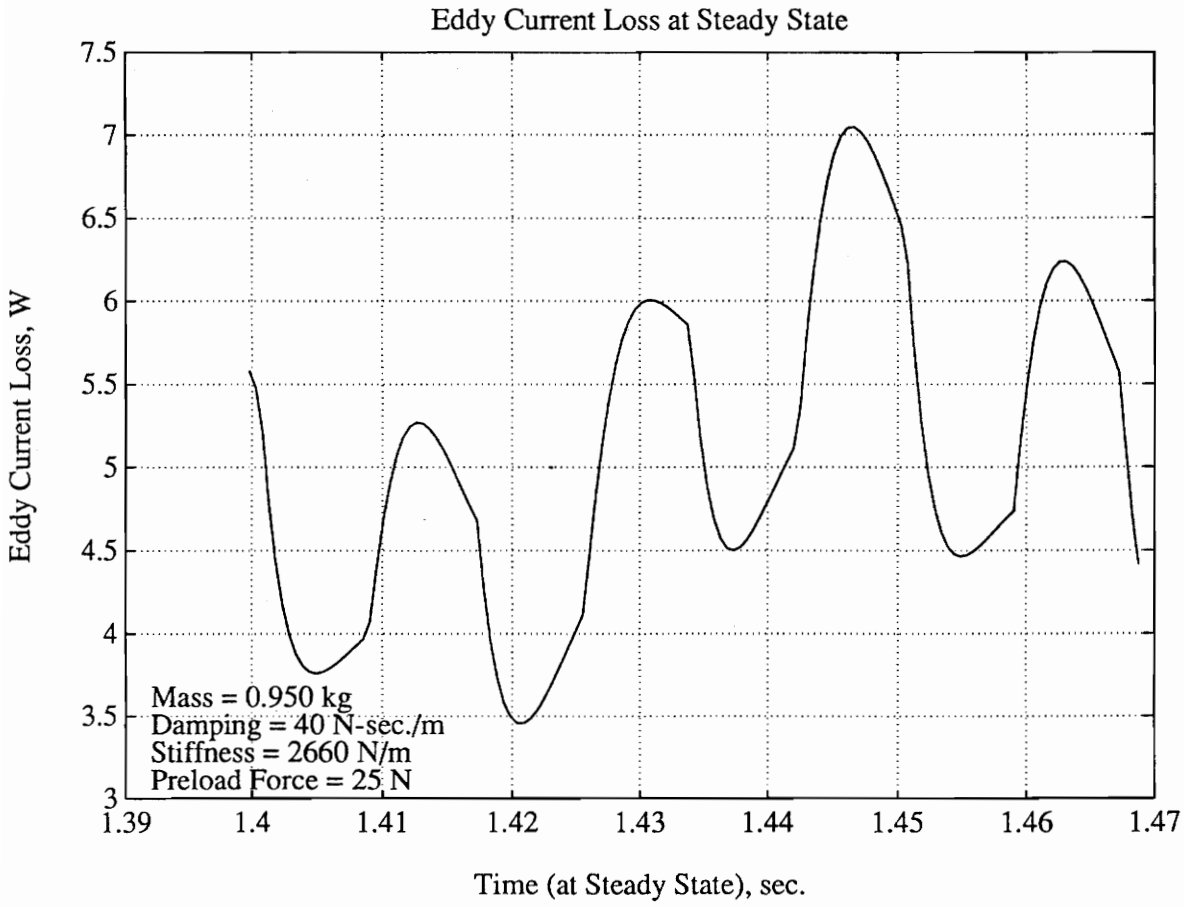


Figure 21. Eddy current loss of motor-reciprocating mechanism at steady state

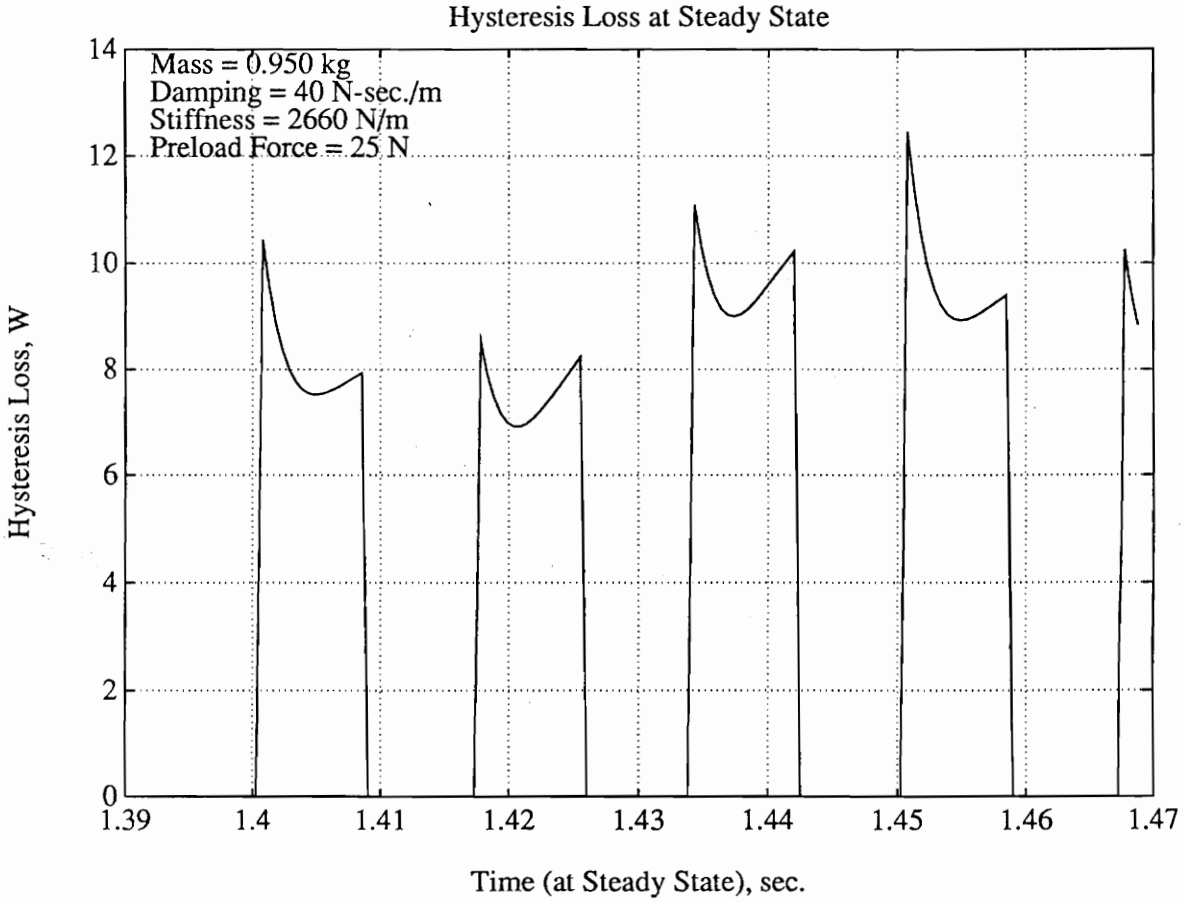


Figure 22. Hysteresis loss of motor-reciprocating mechanism at steady state

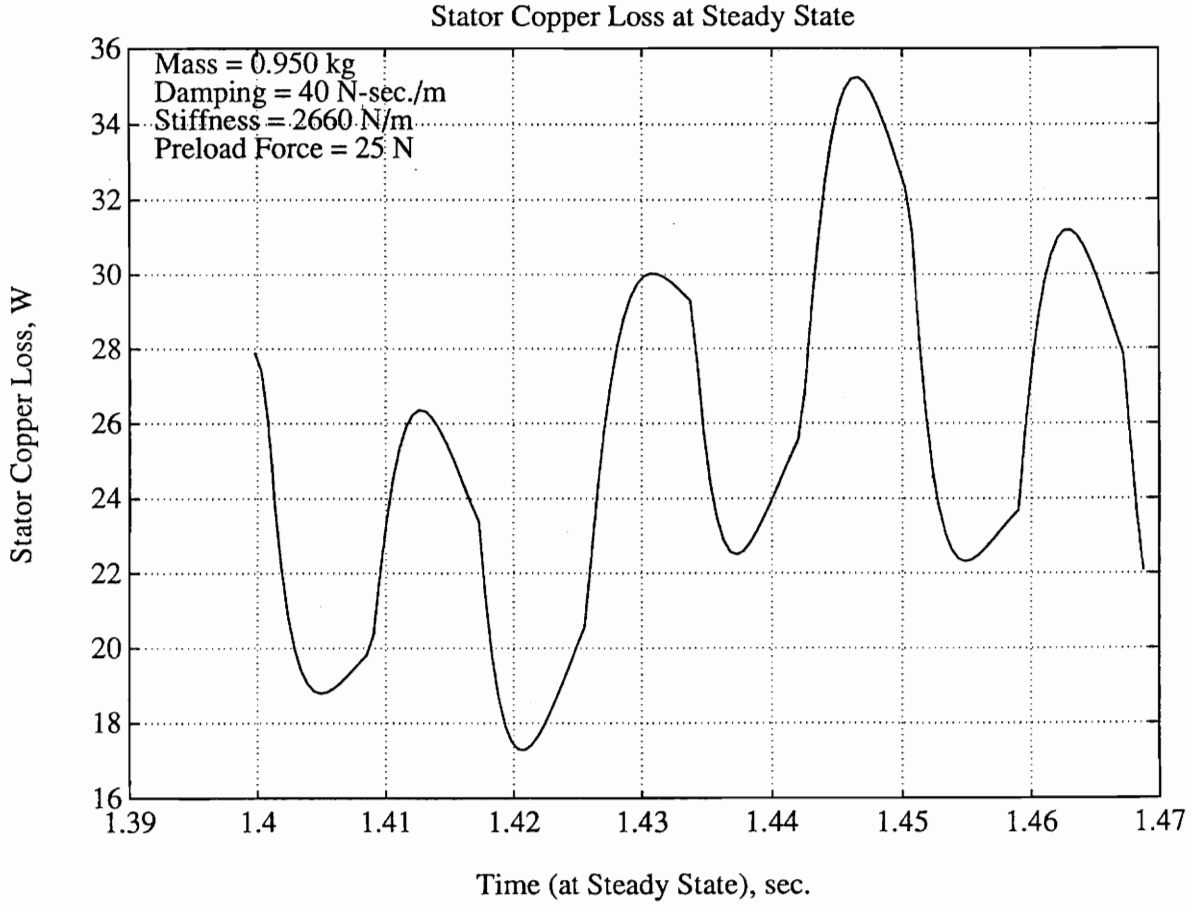


Figure 23. Stator copper loss of motor-reciprocating mechanism at steady state

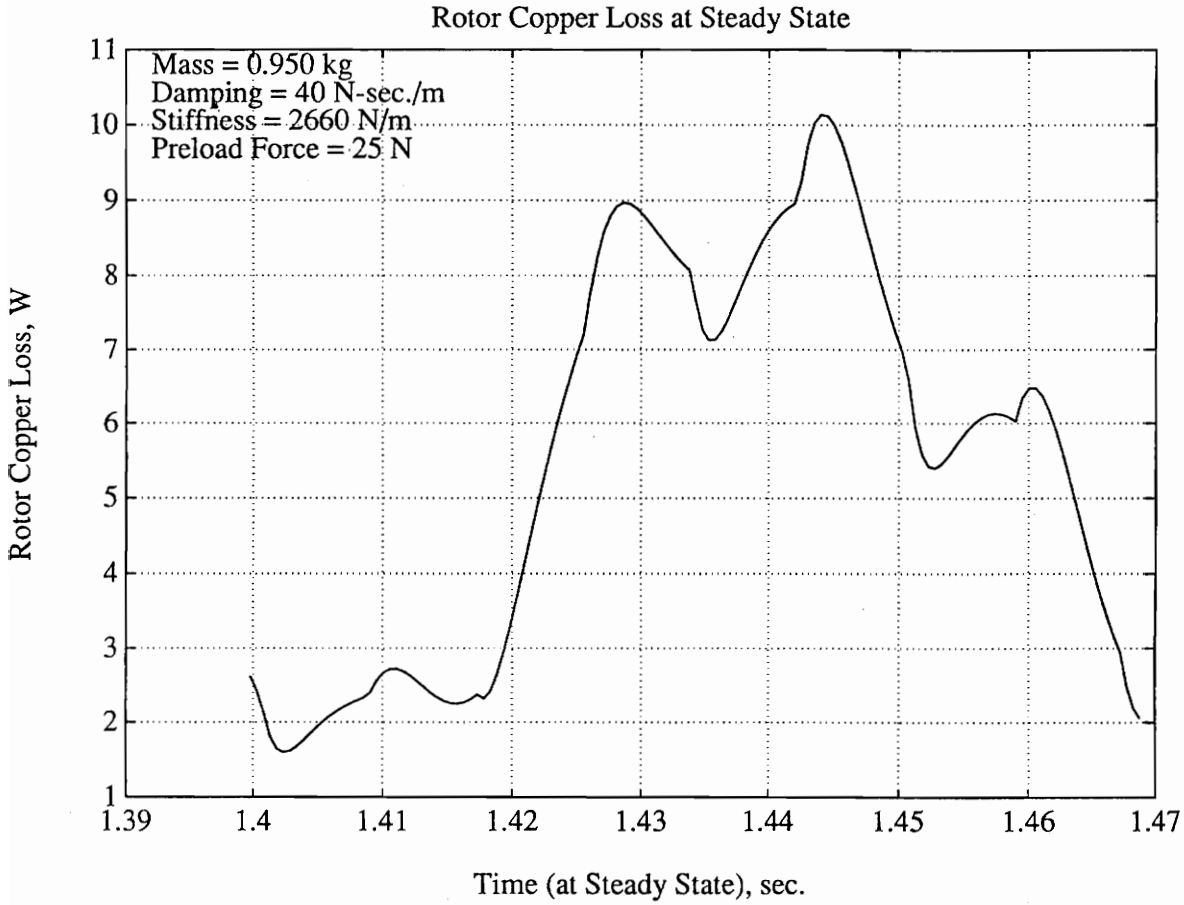


Figure 24. Rotor copper loss of motor-reciprocating mechanism at steady state

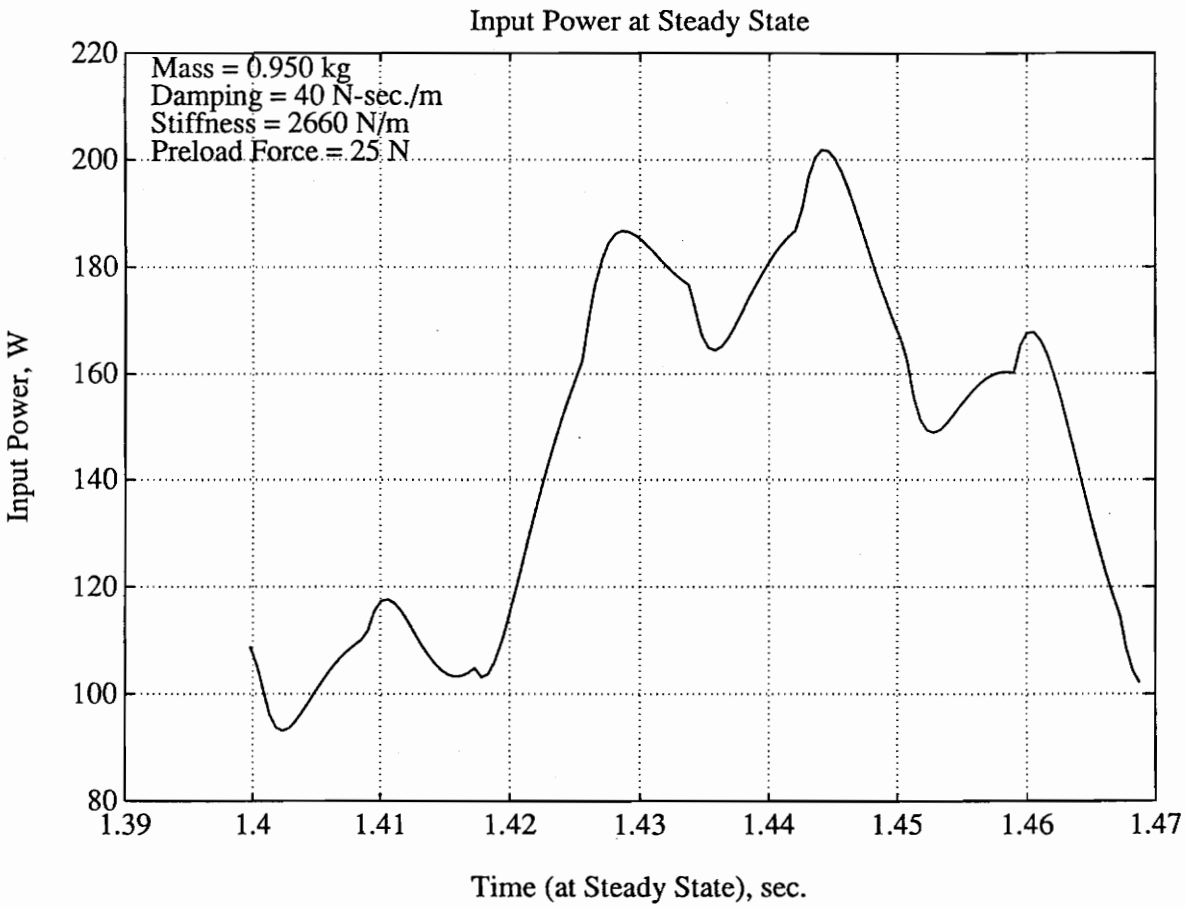


Figure 25. Power input of motor-reciprocating mechanism at steady state

5.4.2 Solution Program

The solution program is used to solve for the reciprocating mechanism parameters which are the lumped mass (M), the viscous damping constant (C), the spring stiffness (K), and the preload force (F_i). The time domain signals generated from the motor simulation program are chosen from one shaft revolution during steady state operation. This process eliminates the leakage problem discussed in the chapter on data processing.

After the time domain signals are properly processed (i.e., selected from a complete cycle at steady state), these signals are analyzed with a discrete Fourier transform (DFT). The frequency spectra show that the low frequency harmonics dominate the motor-mechanism operation (as shown in Chapter 6). These frequency spectra components are used in a least squares solution algorithm to solve the unknown parameters (subroutine LSQRR; used as a least squares solution program) from IMSL[16].

The programs can be referred to in appendix A.

Chapter 6

Solution Technique

6.1 Part One

6.1.1 Harmonically Excited Damped System

This chapter is dedicated to illustrating the effects of the excitation frequency ratio β, noise, sampling rate, and the solution technique by using a simple example. For a reciprocating mass M with a spring, K, and damper, C, all attached as shown in Figure 26, the motion differential equation is

$$M\ddot{x} + C\dot{x} + Kx = f(t) \tag{6.1}$$

where

M = mass(kg)

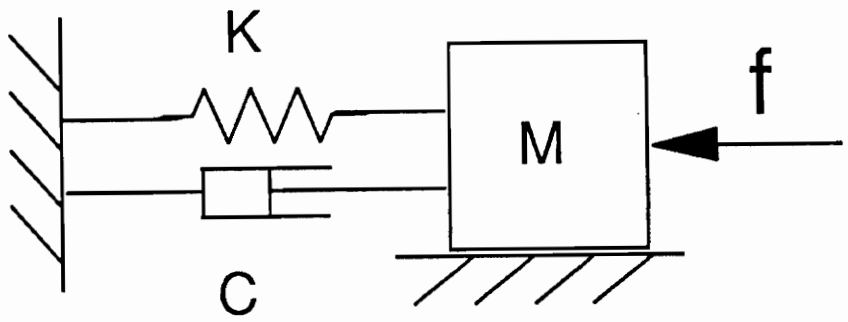


Figure 26. Harmonically excited system of mass attached with damper and spring

C = damping(N-sec./m)

and

K = spring stiffness(N/m)

Suppose that the excitation force $f(t)$ is a harmonic function, say

$$f = F\sin(\omega t) \tag{6.2}$$

then the displacement $x(t)$ can be obtained as shown by Thomson [19] as

$$x = X\sin(\omega t - \phi) \tag{6.3}$$

where displacement amplitude and phase angle are expressed as

$$X = \frac{F}{\sqrt{(K - M\omega^2)^2 + (C\omega)^2}} \tag{6.4}$$

$$\phi = \tan^{-1}\frac{C\omega}{K - M\omega^2} \tag{6.5}$$

The vector relationship of the above solution is shown in Figure 27. It is noted that the phase angle ϕ is a function of ω. For example, if $\omega < \sqrt{K/M}$, $\phi < \frac{\pi}{2}$; if $\omega = \sqrt{K/M}$, $\phi = \frac{\pi}{2}$; if $\omega > \sqrt{K/M}$, $\phi > \frac{\pi}{2}$.

If the excitation force $f(t)$ is a combination of harmonics, the displacement $x(t)$ is the arithmetic sum of the responses resulting from each of these harmonics. For example, the excitation force can be the sum of harmonics of

$$\begin{aligned} f = {} & F_{1b}\sin(\omega t) + F_{1a}\cos(\omega t) + F_{2b}\sin(2\omega t) + F_{2a}\cos(2\omega t) + F_{3b}\sin(3\omega t) + F_{3a}\cos(3\omega t) \\ & + F_{4b}\sin(4\omega t) + F_{4a}\cos(4\omega t) \end{aligned} \tag{6.6}$$

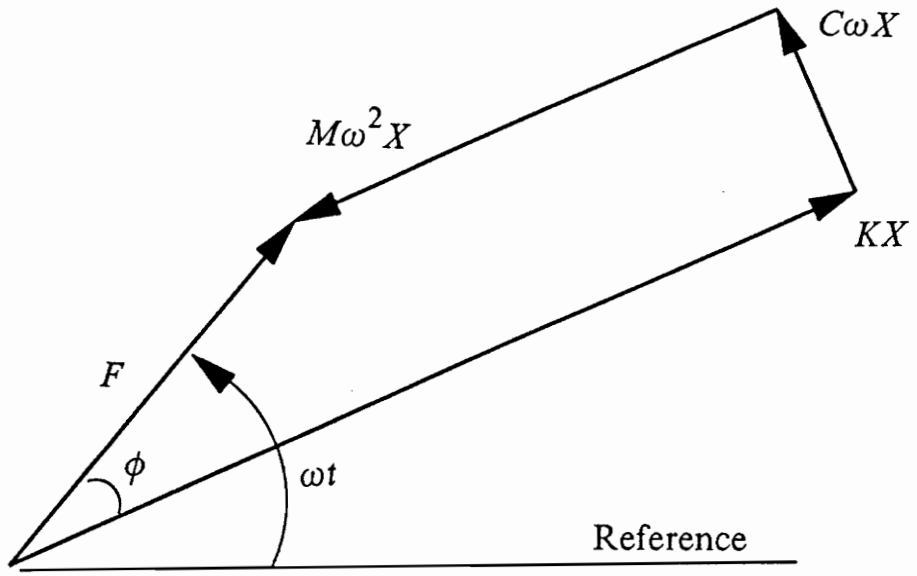


Figure 27. Vector relationship of stiffness force, damping force, mass force, and excitation force (from Thomson)

The excitation force and the parameters used in the demonstration case (shown in Section 6.1.3 "Solution Technique of the Harmonically Excited Damped System") are $F_{1b} = 5, F_{1a} = 10, F_{2b} = 10, F_{2a} = 12, F_{3b} = 9, F_{3a} = -3, F_{4b} = 11, F_{4a} = 7$ N, M = 1 kg, C = 2.5 N-sec./m, K = 400 N/m. It is obvious that the natural frequency of the system is $\omega_n = \sqrt{\frac{K}{M}} = 20$ rad./sec. The frequency spectra of $f(t)$ can be found in Figure 31 of Section 6.1.5 "Effect of Noise".

6.1.2 Least Squares Algorithm

The least squares method is a means used to find the coefficients of a fitted polynomial when the number of available equations (or descriptive functions) exceeds the number of variables to be defined. Consider the polynomial

$$y(x) = a + bx + cx^2 + dx^3 \tag{6.7}$$

and some given points (n, for example, would be (x_1, y_1), (x_2, y_2), (x_3, y_3),...(x_n, y_n)). The sum of the squares of the vertical distances of the points from the polynomial is

$$q = \sum_{j=1}^{n} (y_j - y(x_j))^2 = \sum_{j=1}^{n} (y_j - a - bx_j - cx_j^2 - dx_j^3)^2 \tag{6.8}$$

Since the values of x_j, y_j are known values, the sum of the squares is a function of a, b, c, and d:

$$q = q(a, b, c, d) \tag{6.9}$$

The necessary conditions for the value of q to be minimum are

$$\frac{\partial q}{\partial a} = 0 \tag{6.10}$$

$$\frac{\partial q}{\partial b} = 0 \tag{6.11}$$

$$\frac{\partial q}{\partial c} = 0 \tag{6.12}$$

$$\frac{\partial q}{\partial d} = 0 \tag{6.13}$$

Therefore, the above equations (Equations 6.10 through 6.13) become

$$an + b\sum_{j=1}^{n} x_j + c\sum_{j=1}^{n} x_j^2 + d\sum_{j=1}^{n} x_j^3 = \sum_{j=1}^{n} y_j \tag{6.14}$$

$$a\sum_{j=1}^{n} x_j + b\sum_{j=1}^{n} x_j^2 + c\sum_{j=1}^{n} x_j^3 + d\sum_{j=1}^{n} x_j^4 = \sum_{j=1}^{n} x_j y_j \tag{6.15}$$

$$a\sum_{j=1}^{n} x_j^2 + b\sum_{j=1}^{n} x_j^3 + c\sum_{j=1}^{n} x_j^4 + d\sum_{j=1}^{n} x_j^5 = \sum_{j=1}^{n} x_j^2 y_j \tag{6.16}$$

$$a\sum_{j=1}^{n} x_j^3 + b\sum_{j=1}^{n} x_j^4 + c\sum_{j=1}^{n} x_j^5 + d\sum_{j=1}^{n} x_j^6 = \sum_{j=1}^{n} x_j^3 y_j \tag{6.17}$$

Because the the transformed motion equation (Equation 6.37) (shown in Section 6.2.3 "Solution for Mechanism Parameters") has the same form as the above equations (Equations 6.14 through 6.17), the unknowns M, C, K, and F_i can be determined.

6.1.3 Solution Technique of the Harmonically Excited Damped System

The example shown in Section 6.1.1 "Harmonically Excited Damped System" can be used to solve for the parameters of mass and spring stiffness in the single degree of freedom system. The known values of mass and spring stiffness are 1 kg and 400 N/m, respectively. The discussion of damping (2.5 N-sec./m) will be found in Part Two at the end of this chapter.

The Fourier transform of $\ddot{x}(t)$ is defined as the following symbol:

$$F\{\ddot{x}(t)\} = \ddot{\bar{x}} \tag{6.18}$$

The time domain acceleration $\ddot{x}(t)$ can be expressed as the sum of the harmonics as

$$\ddot{x}(t) = \ddot{x}_1(t) + \ddot{x}_2(t) + ... + \ddot{x}_n(t) \tag{6.19}$$

Therefore,

$$\begin{aligned} F\{\ddot{x}\} &= F\{\ddot{x}_1 + \ddot{x}_2 + ... + \ddot{x}_n\} \\ &= F\{\ddot{x}_1\} + F\{\ddot{x}_2\} + ... + F\{\ddot{x}_n\} \\ &= \ddot{\bar{x}}_1 + \ddot{\bar{x}}_2 + ... + \ddot{\bar{x}}_n \end{aligned} \tag{6.20}$$

The Fourier transform of $\ddot{x}_1$, $\ddot{\bar{x}}_1$, the first harmonic of acceleration in the frequency domain, is expressed as the sum of the real part and imaginary parts.

$$\ddot{\bar{x}}_1 = \ddot{\bar{x}}_{1c} + i\ddot{\bar{x}}_{1s} \tag{6.21}$$

Recall that the vector relationship as shown in Figure 27, the excitation force $f(t)$ equals the sum of the stiffness force, KX, damping force, $C\omega X$, and mass force, $M\omega^2 X$. Therefore, the sine component of $f(t)$ equals the sum of the sine components of the stiffness, damping, and mass forces.

So does the $f(t)$ cosine component equal the sum of the cosine components of the stiffness, damping, and mass forces.

Considering that there are four harmonics in the excitation force, eight equations (four cosine components and four sine components) can be used to solve the unknown parameters. For example, if the cosine component of the first harmonic of the excitation force $f(t)$ in the frequency domain is 10 N, then we may find the corresponding terms of acceleration, velocity, and change of displacement (also the cosine component of the first harmonic of these variables). Thus we can form an equation as

$$M\ddot{\bar{x}}_{1c} + C\dot{\bar{x}}_{1c} + K\bar{x}_{1c} = \bar{f}_{1c} \tag{6.22}$$

where

$\ddot{\bar{x}}_{1c}$ = cosine component (real part) of the first harmonic of acceleration in frequency domain

and

$\dot{\bar{x}}_{1c}$ = cosine component (real part) of the first harmonic of the velocity in frequency domain

$\bar{x}_{1c}$ = cosine component of the first harmonic of the change of displacement in frequency domain

Similarly, we can obtain the other equations by choosing different cosine or sine components of the dominating harmonics (in this case, the dominating harmonics means the first, second, third, and fourth harmonics). Therefore, the matrix form of motion equations in the frequency domain result as

$$\begin{bmatrix} \ddot{\bar{x}}_{1c} & \dot{\bar{x}}_{1c} & \bar{x}_{1c} \\ \ddot{\bar{x}}_{1s} & \dot{\bar{x}}_{1s} & \bar{x}_{1s} \\ \ddot{\bar{x}}_{2c} & \dot{\bar{x}}_{2c} & \bar{x}_{2c} \\ \ddot{\bar{x}}_{2s} & \dot{\bar{x}}_{2s} & \bar{x}_{2s} \\ \ddot{\bar{x}}_{3c} & \dot{\bar{x}}_{3c} & \bar{x}_{3c} \\ \ddot{\bar{x}}_{3s} & \dot{\bar{x}}_{3s} & \bar{x}_{3s} \\ \ddot{\bar{x}}_{4c} & \dot{\bar{x}}_{4c} & \bar{x}_{4c} \\ \ddot{\bar{x}}_{4s} & \dot{\bar{x}}_{4s} & \bar{x}_{4s} \end{bmatrix} \begin{bmatrix} M \\ C \\ K \end{bmatrix} = \begin{bmatrix} \bar{f}_{1c} \\ \bar{f}_{1s} \\ \bar{f}_{2c} \\ \bar{f}_{2s} \\ \bar{f}_{3c} \\ \bar{f}_{3s} \\ \bar{f}_{4c} \\ \bar{f}_{4s} \end{bmatrix} \tag{6.23}$$

where

$\bar{f}_{ns}$ = sine component (imaginary part) of the n'th harmonic of the excitation force in frequency domain

$\bar{f}_{nc}$ = cosine component of the n'th harmonic of the excitation force in frequency domain

$\ddot{\bar{x}}_{ns}$ = sine component of the n'th harmonic of the acceleration in frequency domain

$\ddot{\bar{x}}_{nc}$ = cosine component of the n'th harmonic of the acceleration in frequency domain

$\dot{\bar{x}}_{ns}$ = sine component of the n'th harmonic of the velocity in frequency domain

$\dot{x}_{nc}$ = cosine component of the n'th harmonic of the velocity in frequency domain

$\bar{x}_{ns}$ = sine component of the n'th harmonic of the change of displacement in frequency domain

$\bar{x}_{nc}$ = cosine component of the n'th harmonic of the change of displacement in frequency domain

By solving the above set of equations (Equation 6.23), the values of M, C, and K can be obtained. It is also found that if the number of equations is less than eight (e.g. seven, six), the accuracy of the result is diminished. According to different situations (different excitation frequency and signal to noise ratio), the values of M, C, and K are listed in the following sections.

6.1.4 Effect of the Excitation Frequency Ratio

As the excitation frequency ratio $\beta, \frac{\omega}{\omega_n}$, is varied from 0.65 through 5.0, it is found that excitation frequency ratio β, will not change the accuracies of the calculation of the mass and the spring stiffness. These results are shown in Table 1 and Figure 28.

6.1.5 Effect of Random Noise

Figures 30 through 41 show the excitation forces with random noises in both the time domain and frequency domain. As the SNR, the signal-to-noise ratio, defined as the ratio of the noise variance to the square of the maximum signal value, is varied from 0 through 5.0, the accuracies of the calculated parameters are also changed (as shown in Figure 29). Figures 30 through 41 show that the larger the noise, the larger the calculated relative error.

Table 1. **Mass and stiffness errors without noise**

$\beta(=\frac{\omega}{\omega_n})$		Mass (kg)	Stiffness (N/m)
0.65	True	1.00000	400.00000
	Calculated	0.99999	399.99951
	Error %	-0.00100	-0.00010
0.80	True	1.00000	400.00000
	Calculated	0.99999	399.99805
	Error %	-0.00100	-0.00049
0.95	True	1.00000	400.00000
	Calculated	0.99999	399.99731
	Error %	-0.00100	-0.00070
1.05	True	1.00000	400.00000
	Calculated	0.99998	399.99292
	Error %	-0.00200	-0.00180
1.20	True	1.00000	400.00000
	Calculated	0.99999	400.00415
	Error %	-0.00100	0.00100
1.50	True	1.00000	400.00000
	Calculated	1.00000	400.00684
	Error %	0.00000	0.00171
2.00	True	1.00000	400.00000
	Calculated	1.00000	399.99976
	Error %	0.00000	-0.00006
2.50	True	1.00000	400.00000
	Calculated	1.00000	399.99976
	Error %	0.00000	-0.00006
3.00	True	1.00000	400.00000
	Calculated	1.00000	400.00317
	Error %	0.00000	0.00079
3.50	True	1.00000	400.00000
	Calculated	1.00000	399.99854
	Error %	0.00000	-0.00037

Table 2. Results of calculated mass(M) and stiffness(K)

Signal to Noise Ratio		Mass (kg)	Stiffness (N/m)
0	True	1.0000	400.0000
	Calculated	1.0000	399.9995
	Error %	0.0000	-0.0001
0.1	True	1.0000	400.0000
	Calculated	0.9950	397.5130
	Error %	-0.5000	-0.6200
0.3	True	1.0000	400.0000
	Calculated	0.9830	391.7900
	Error %	-1.7000	-2.1000
0.5	True	1.0000	400.0000
	Calculated	0.9700	386.1400
	Error %	-3.0000	-3.5000
0.8	True	1.0000	400.0000
	Calculated	0.9530	377.9250
	Error %	-4.7000	-5.5000
1.0	True	1.0000	400.0000
	Calculated	0.9410	372.6280
	Error %	-5.9000	-6.8000
3.0	True	1.0000	400.0000
	Calculated	0.8400	320.5260
	Error %	-16.0000	-19.9000
5.0	True	1.0000	400.0000
	Calculated	0.7590	281.5980
	Error %	-24.1000	-29.6000

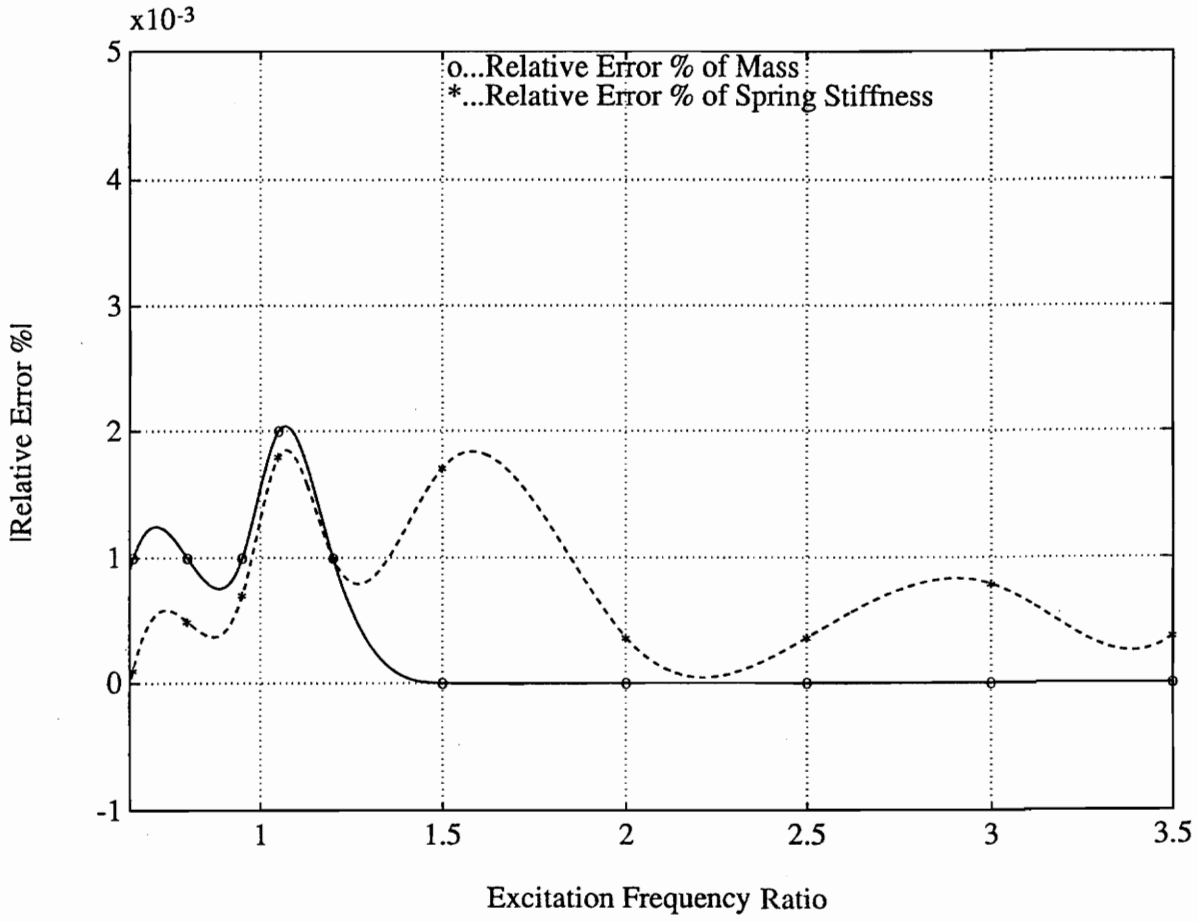


Figure 28. Relative errors % of mass and stiffness vs. excitation frequency ratio

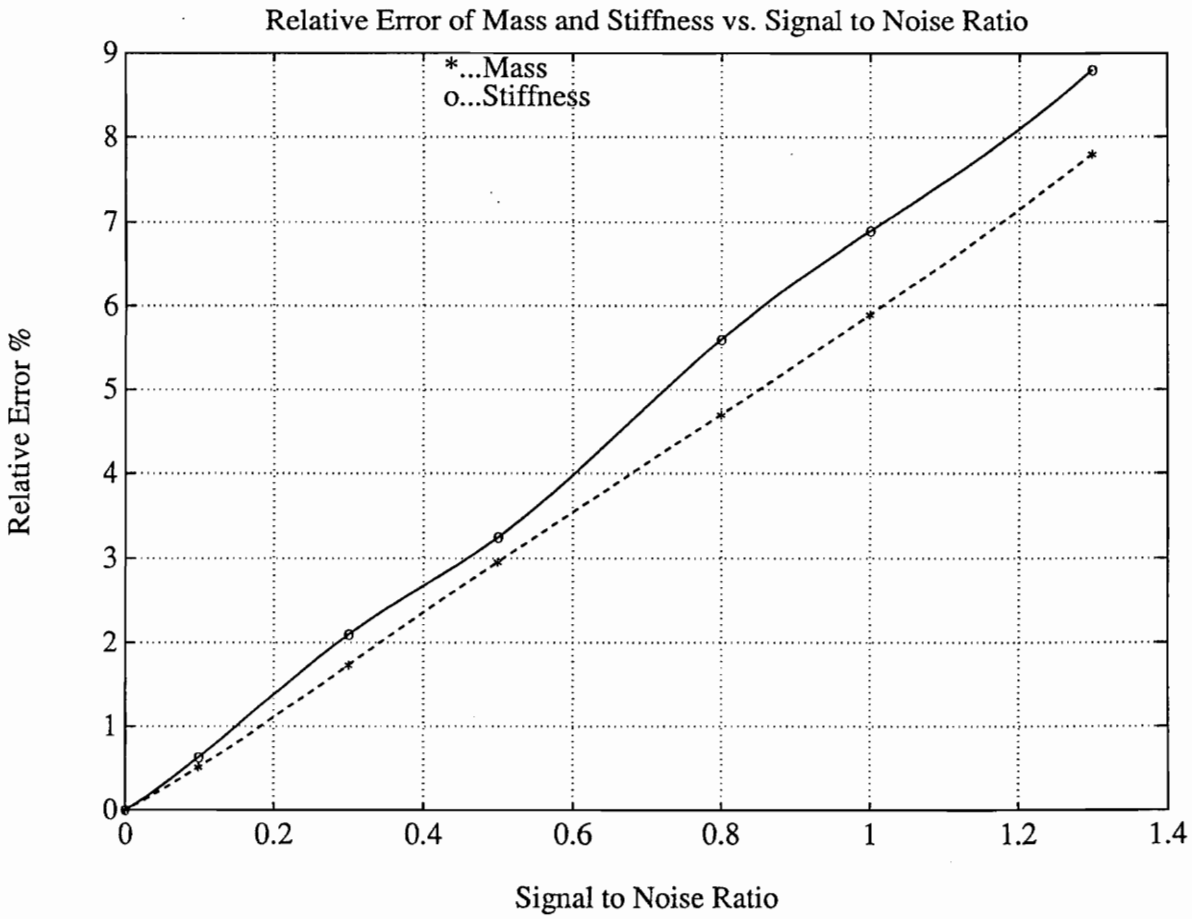


Figure 29. Relative error % of mass and stiffness vs. SNR

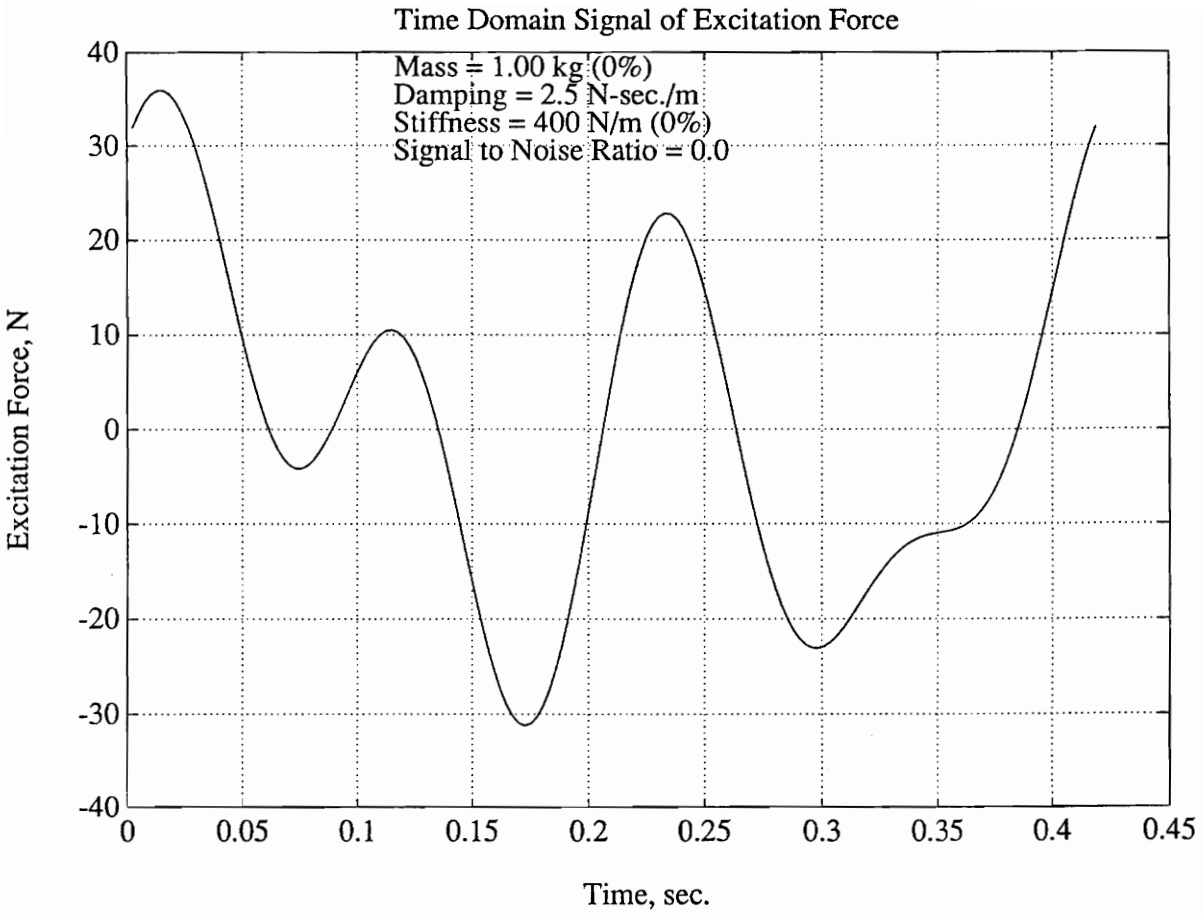


Figure 30. Excitation force with SNR = 0 in time domain

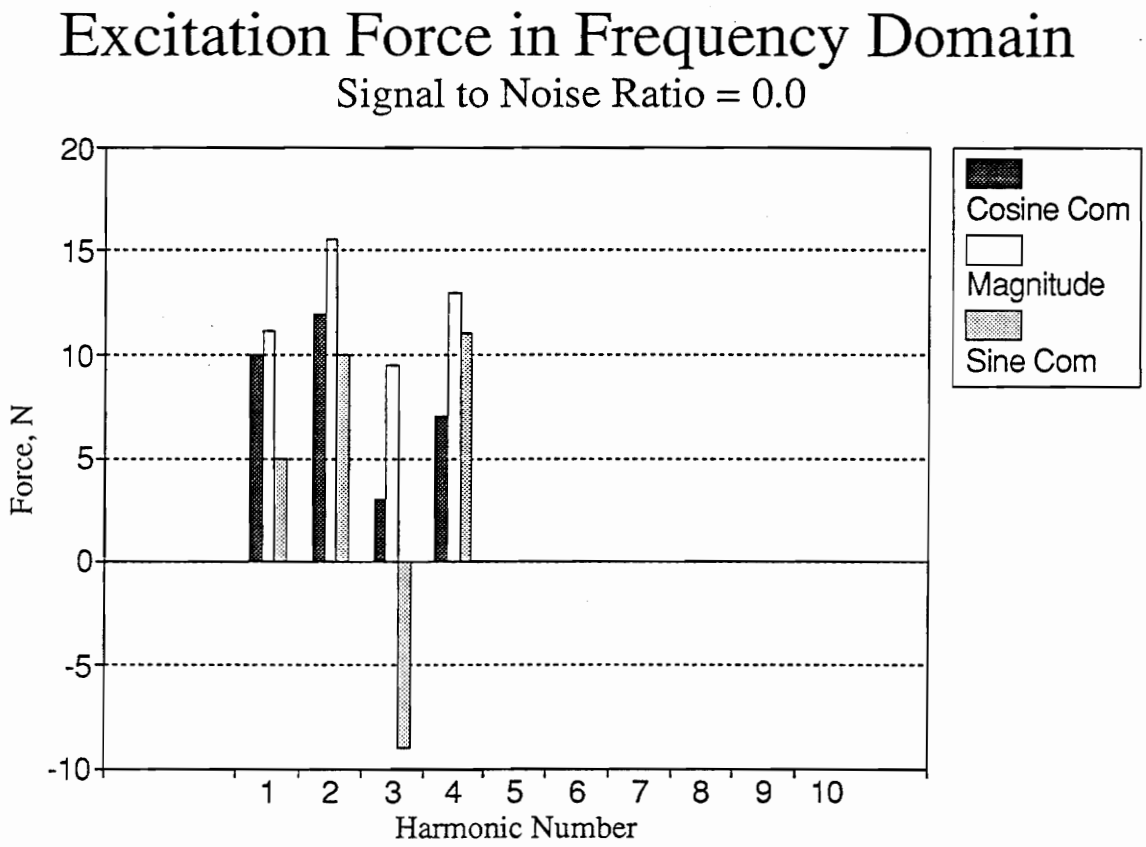


Figure 31. Excitation force with SNR = 0 in frequency domain

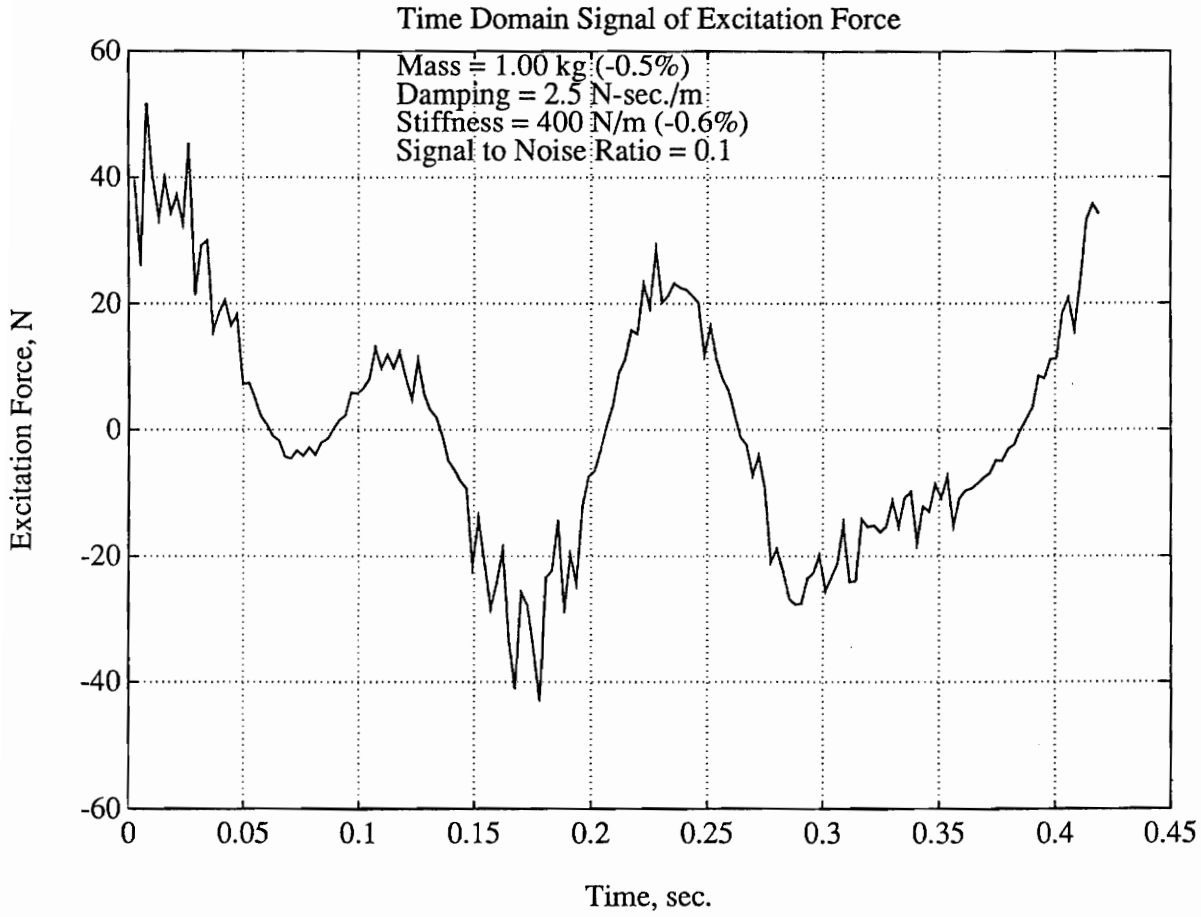


Figure 32. Excitation force with SNR = 0.1 in time domain

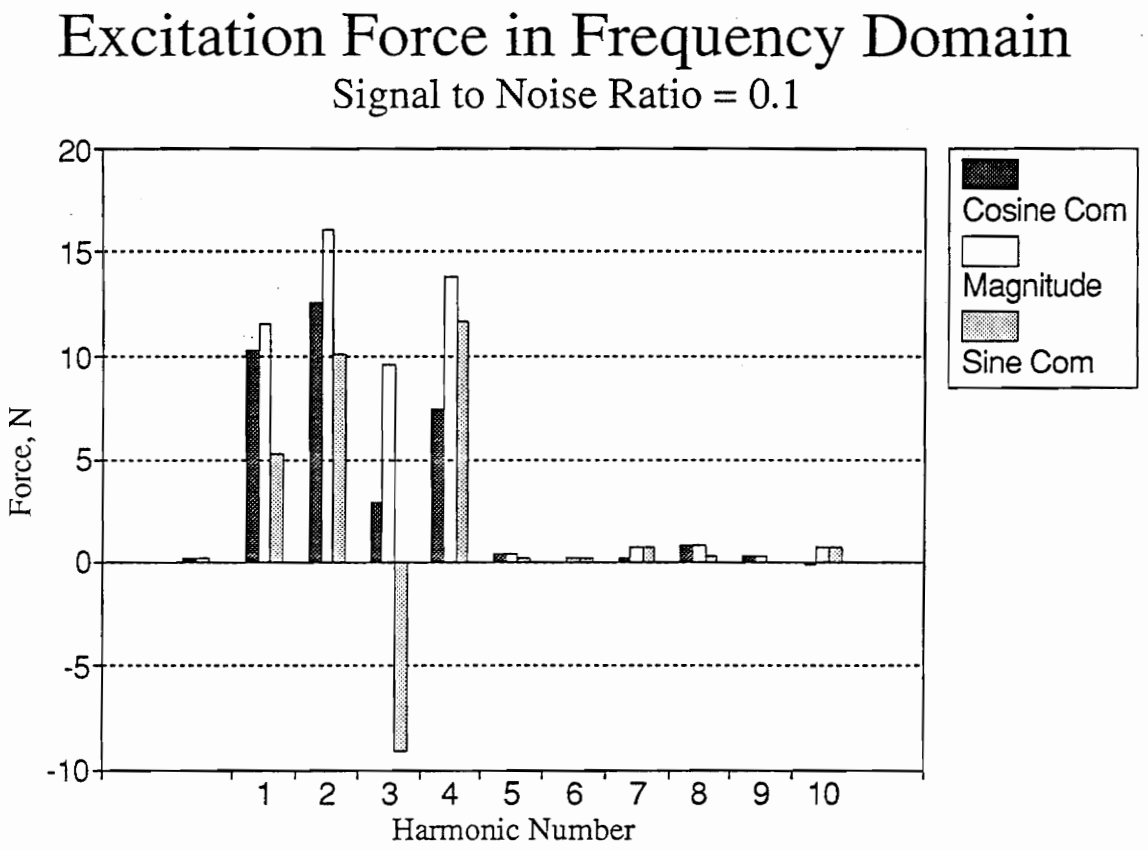


Figure 33. **Excitation force with SNR = 0.1 in frequency domain**

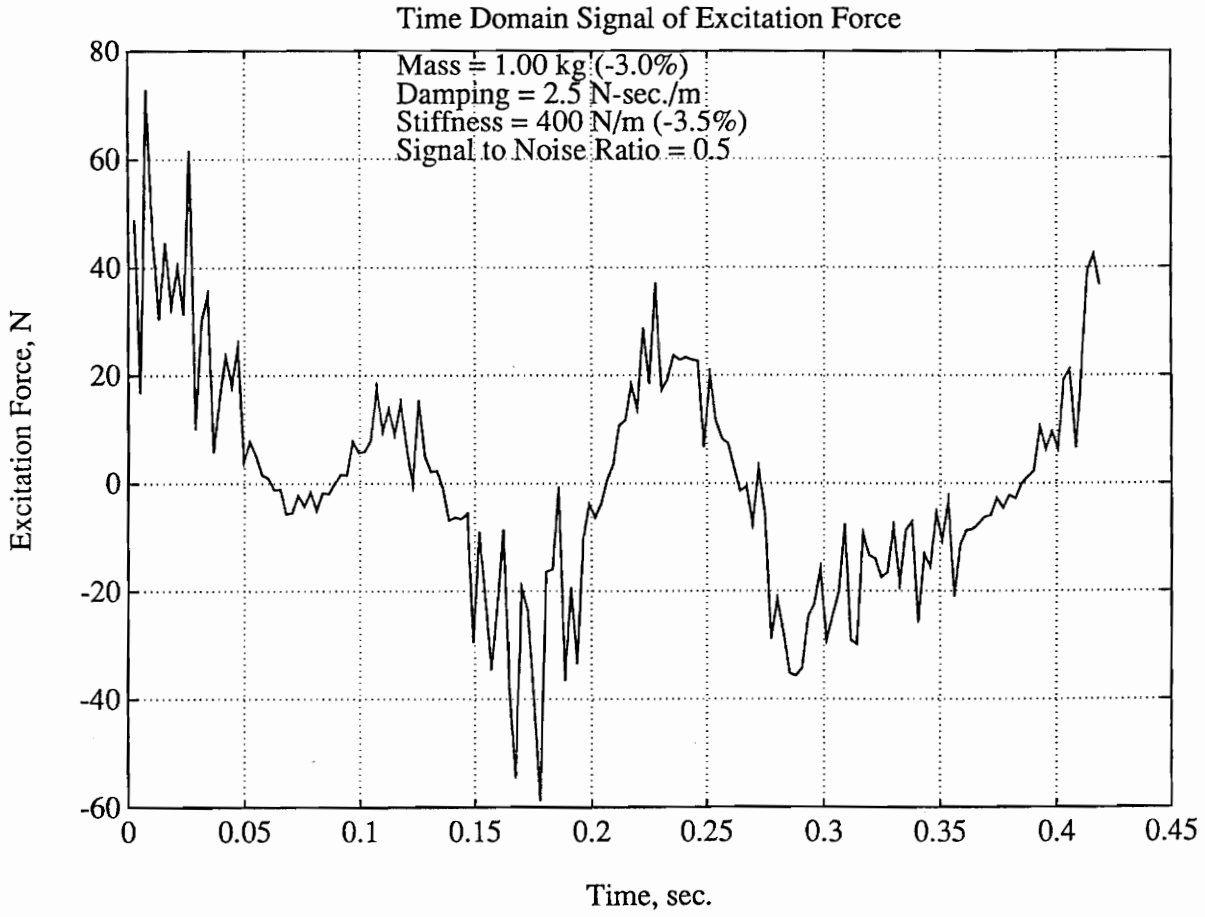


Figure 34. Excitation force with SNR = 0.5 in time domain

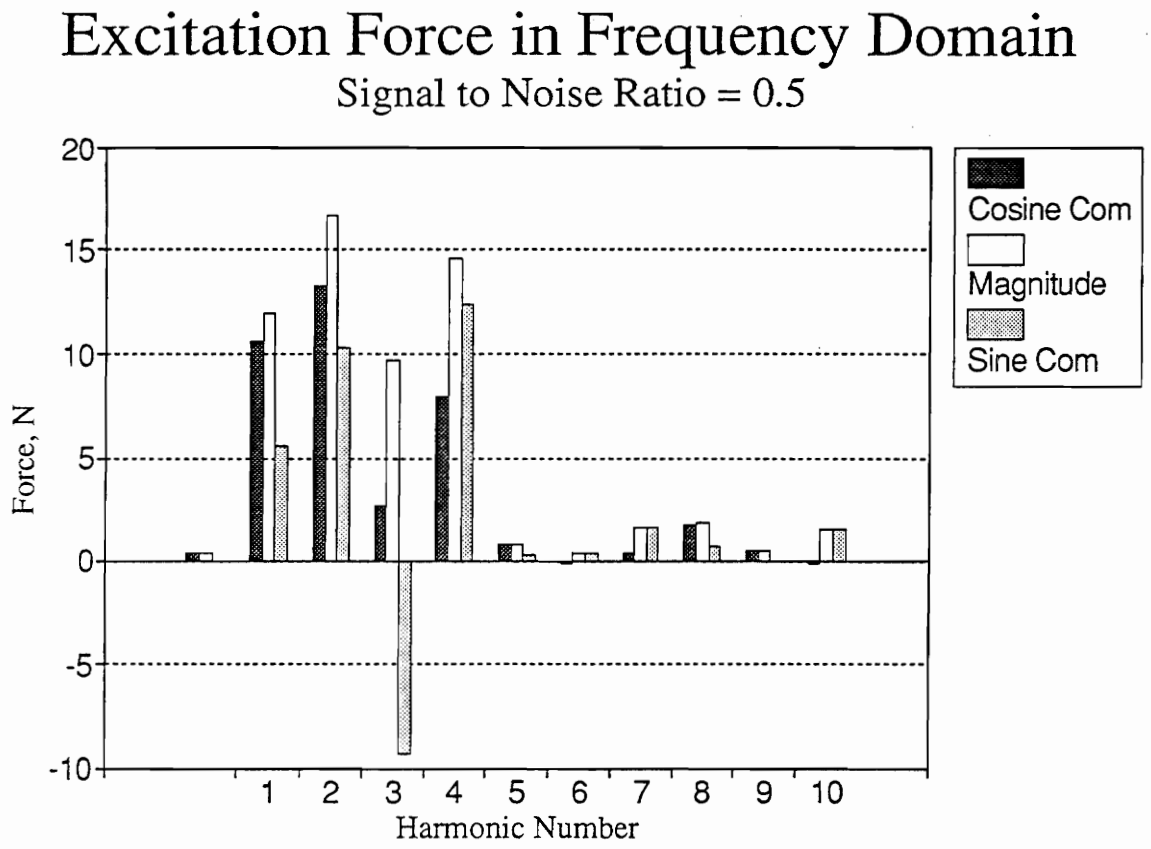


Figure 35. Excitation force with SNR = 0.5 in frequency domain

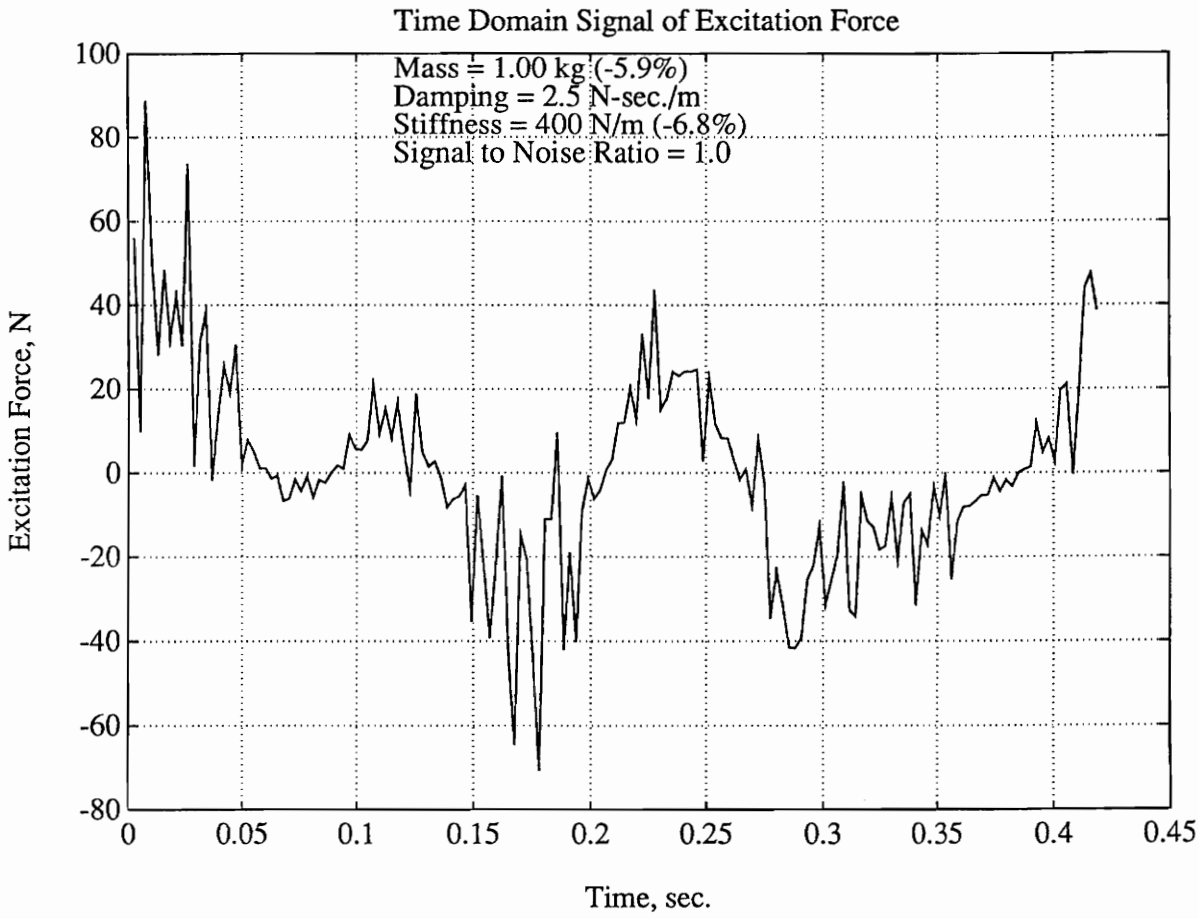


Figure 36. Excitation force with SNR = 1.0 in time domain

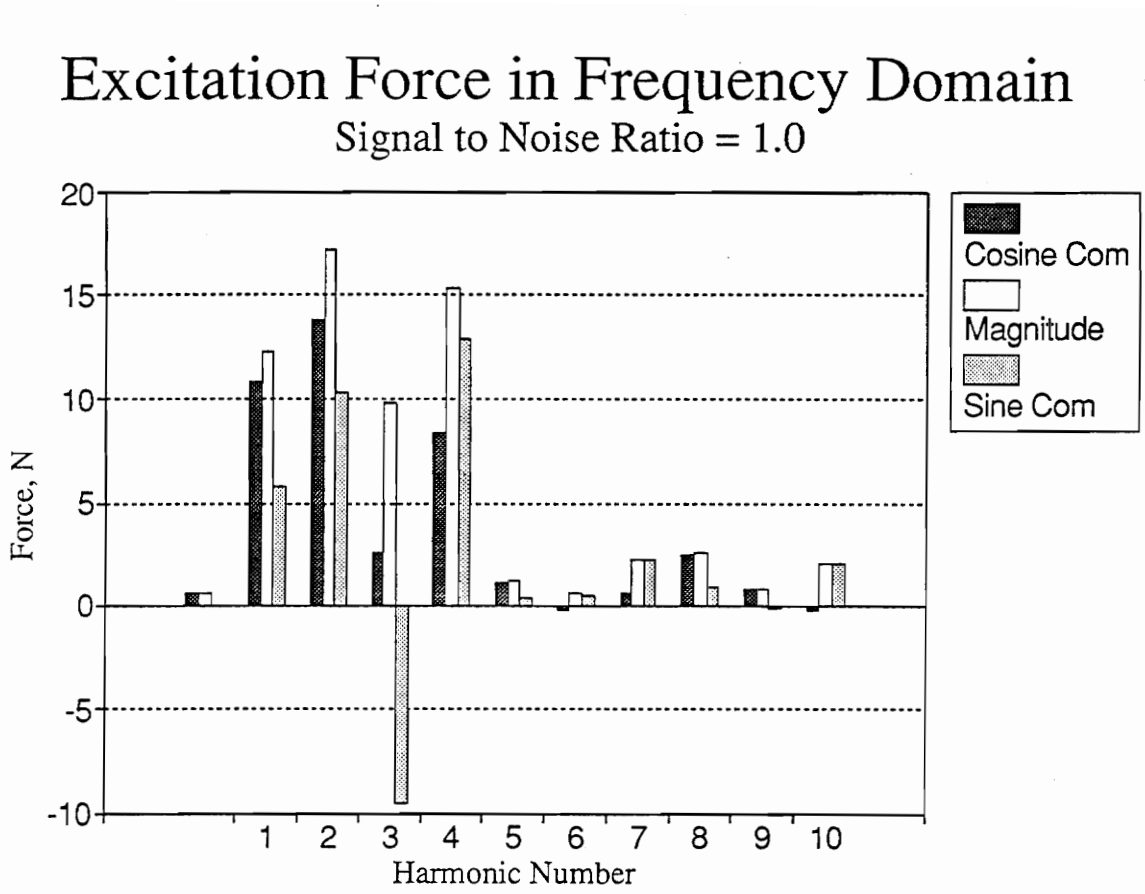


Figure 37. Excitation force with SNR = 1.0 in frequency domain

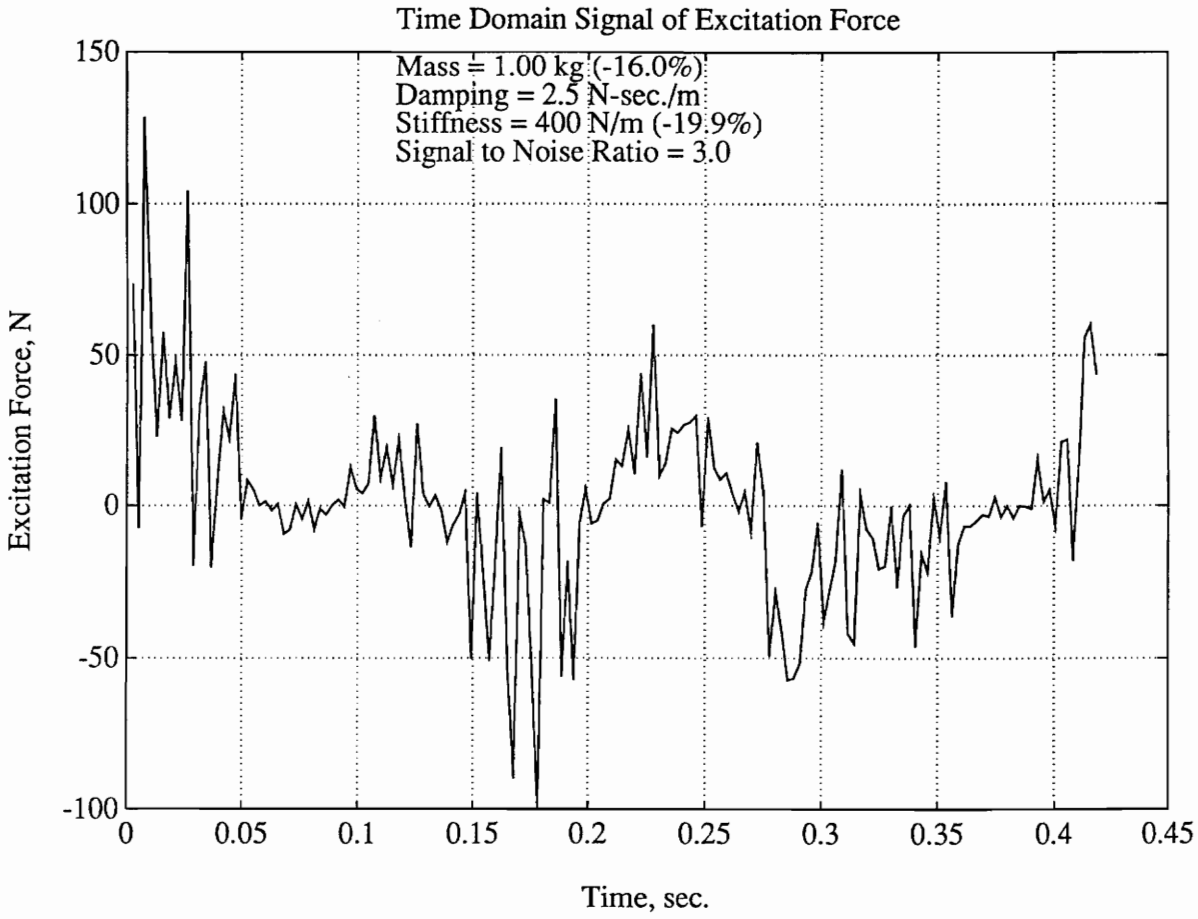


Figure 38. Excitation force with SNR = 3.0 in time domain

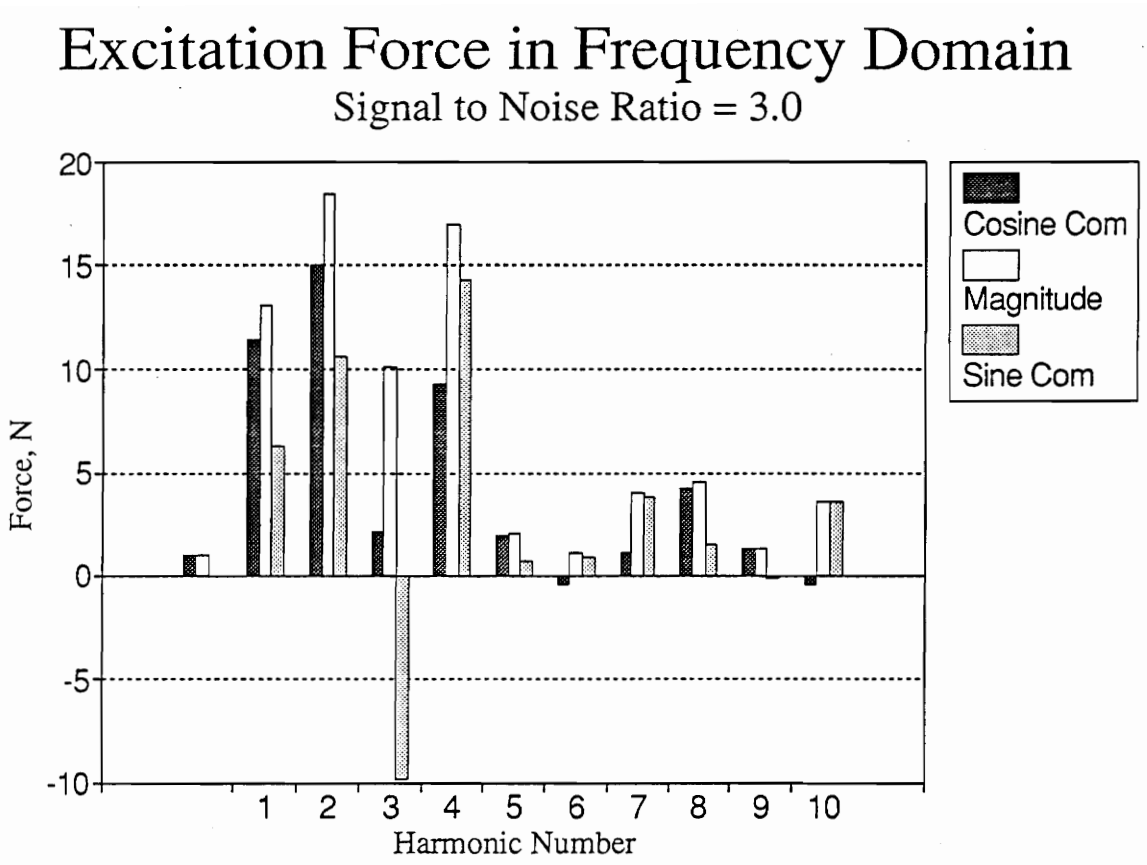


Figure 39. Excitation force with SNR = 3.0 in frequency domain

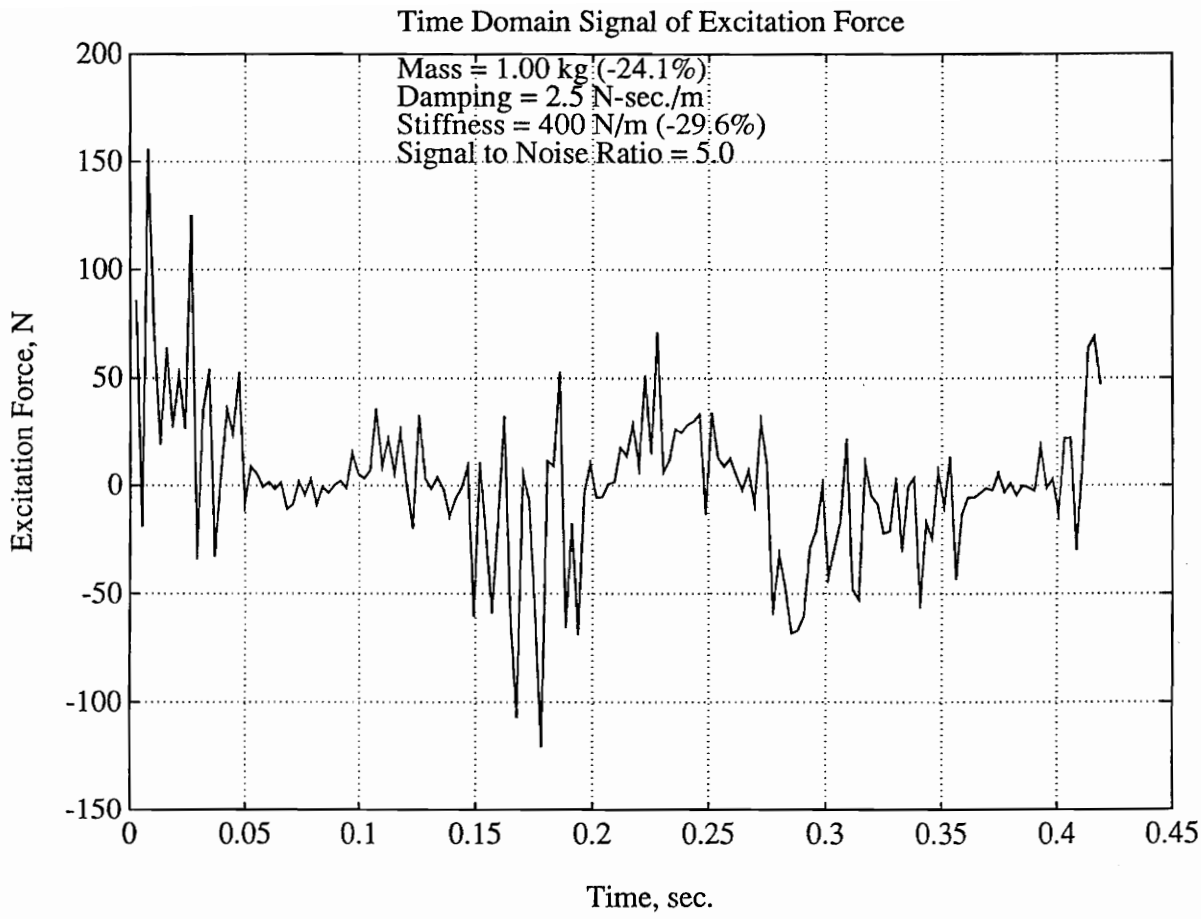


Figure 40. Excitation force with SNR = 5.0 in time domain

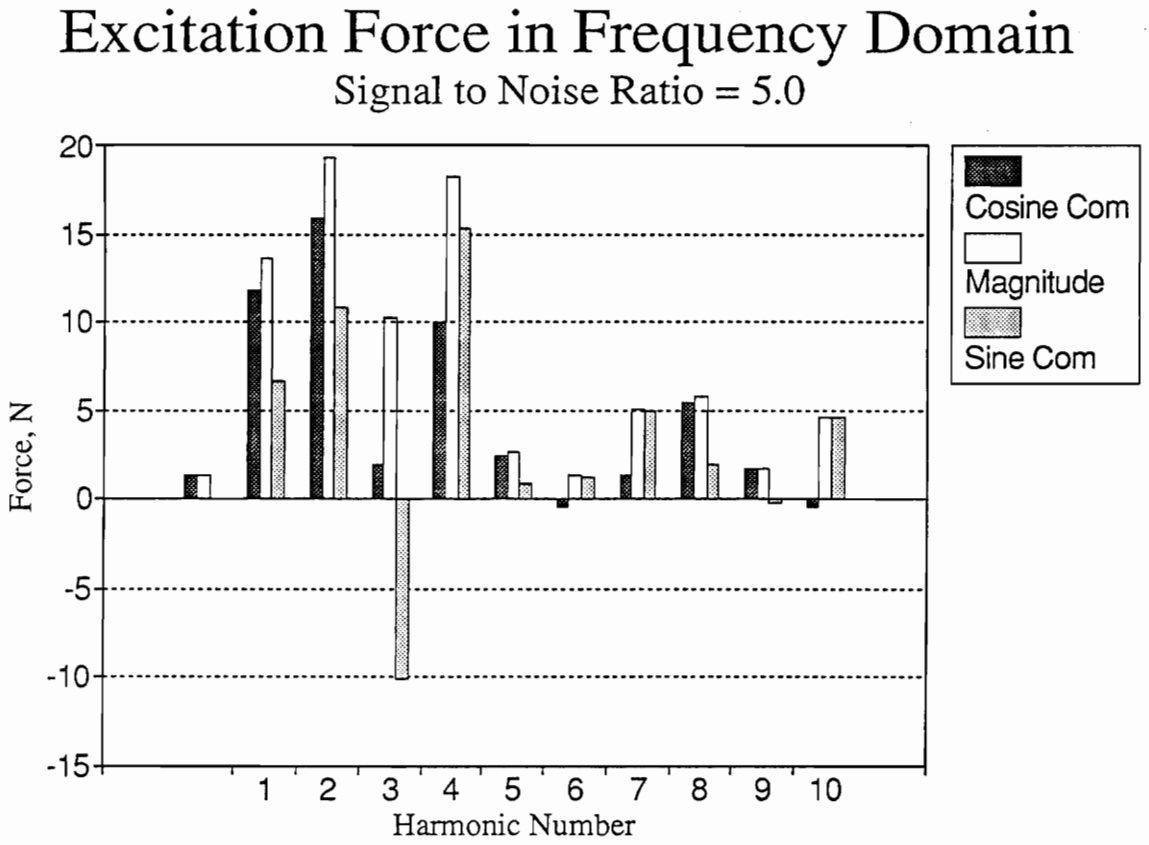


Figure 41. Excitation force with SNR = 5.0 in frequency domain

6.2 *Part Two*

6.2.1 Data Pre-processing

According to the previous data acquisition and simulation description, it is necessary to take a complete cycle of data to avoid leakage. The sampling frequency should be at least two times of the largest frequency of the signals to be sampled, as discussed in Chapter Five. A higher sampling frequency would pick up more higher frequency signals but would also consume more frequency transform calculation time. The sampling rate can be used in the motor simulation program is the maximum sampling rate, 9600 Hz, divided by a positive integer. The sampling rate to be used in these procedures is 1920 Hz; this high sampling rate will require substantial frequency transform calculation time but can capture higher frequency signals.

From the load torque analysis in Chapter Two, the load torque signal can be divided into the contributions of mass, damping, stiffness, and preload force. These load torque contributions are determined by different dynamic variables except the preload force. The plots of the piston acceleration, velocity, and displacement can be referred to in Figure 42 through Figure 45. The mass load torque contribution is obtained by the multiplication of the lumped reciprocating mass and the piston acceleration. The damping contribution is from the product of the damping constant and the piston velocity. The stiffness contribution is from the spring stiffness multiplied by the piston displacement difference. The contribution of preload force is from the multiplication of preload force and a position function as shown on the section of Load Torque Separation.

After the time-domain data is generated, it is necessary to choose the data of a cycle from $\theta = 0$ to $\theta = 2\pi$ at steady state. Then the data is transformed into the frequency domain through a discrete Fourier transformation (DFT) process. Then the frequency spectrum of each signal is used to solve for the unknown parameters through a least squares algorithm. In the case of three unknowns,

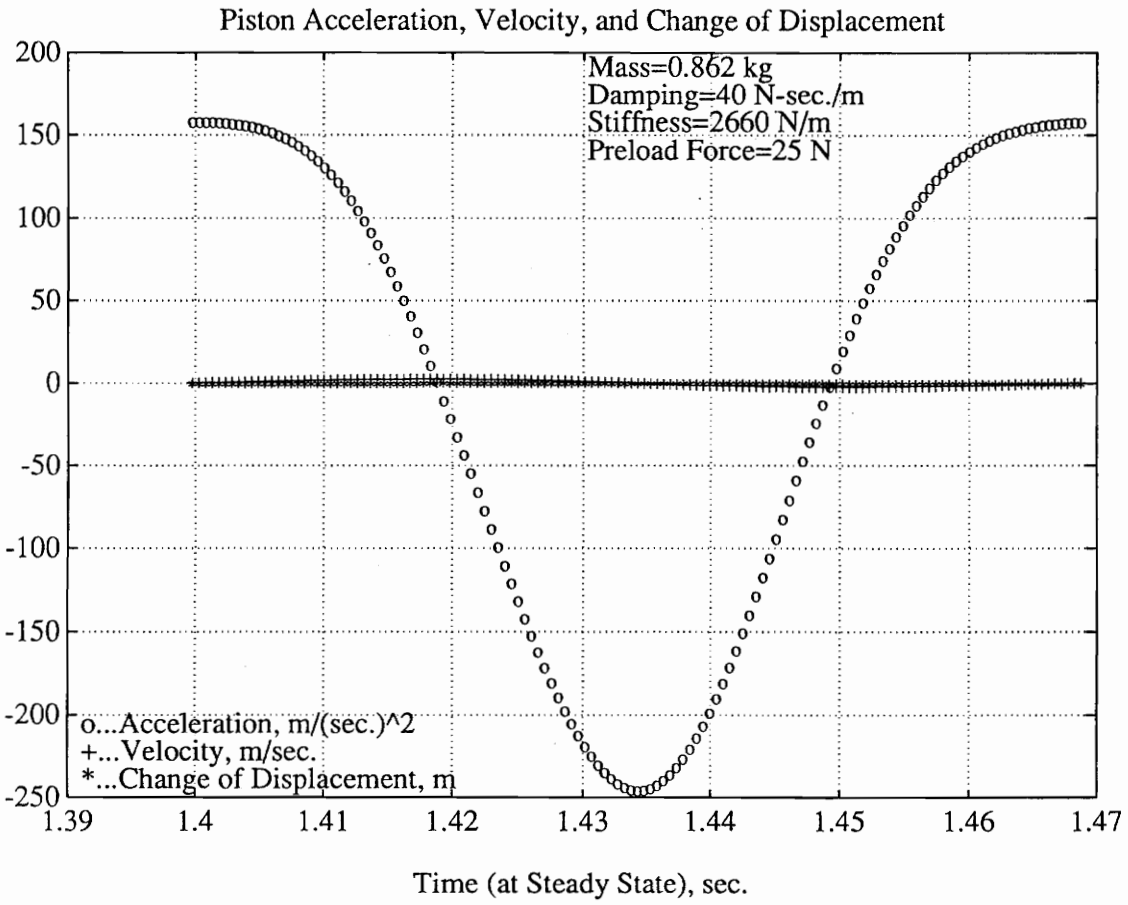


Figure 42. Piston acceleration, velocity, and change of displacement

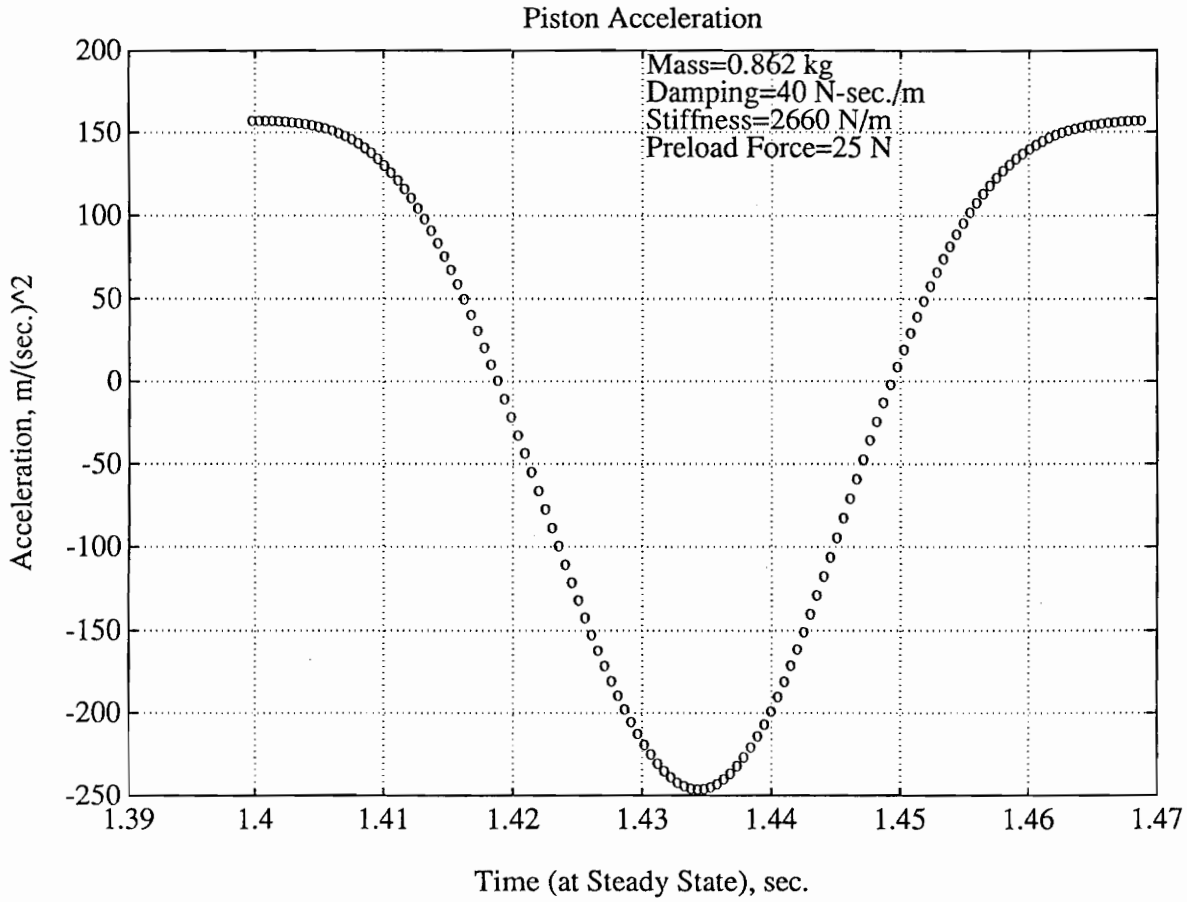


Figure 43. Piston acceleration

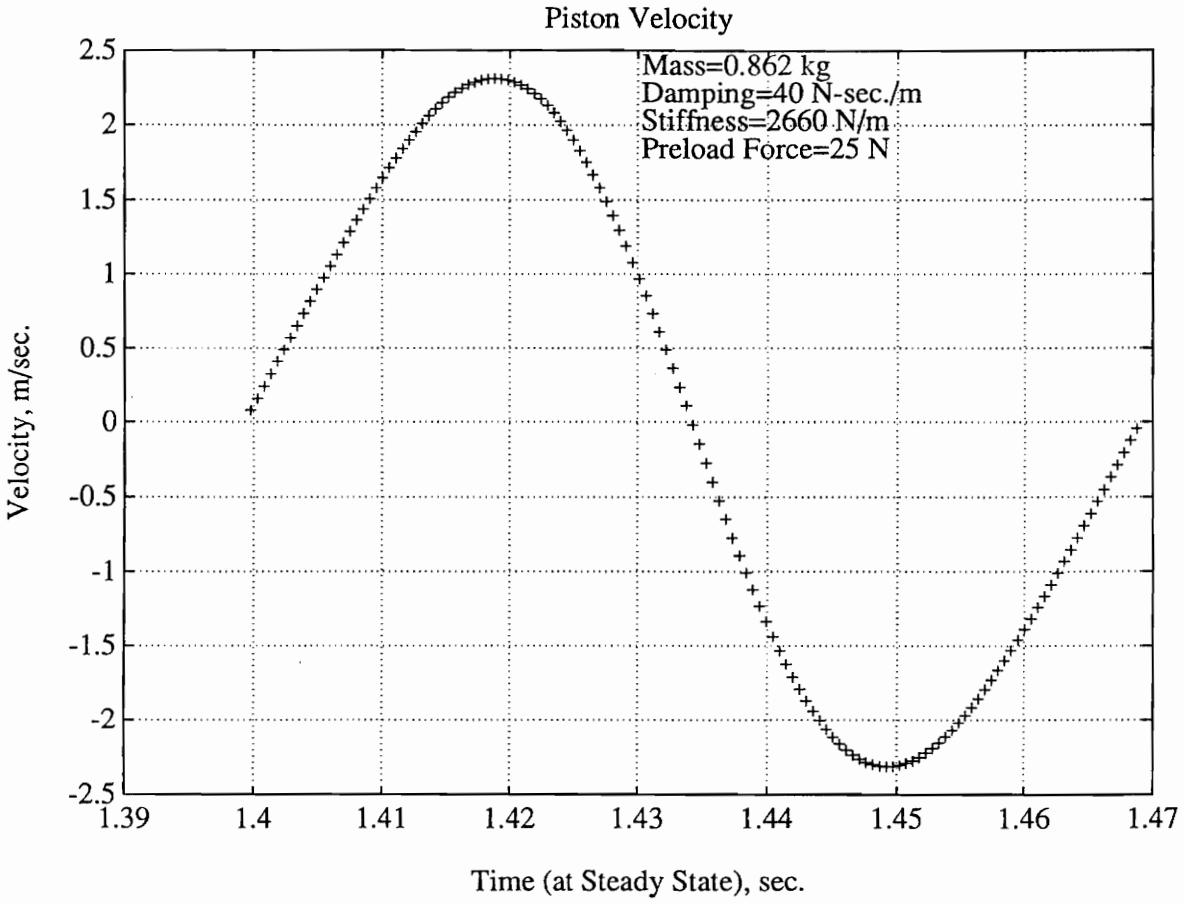


Figure 44. Piston velocity

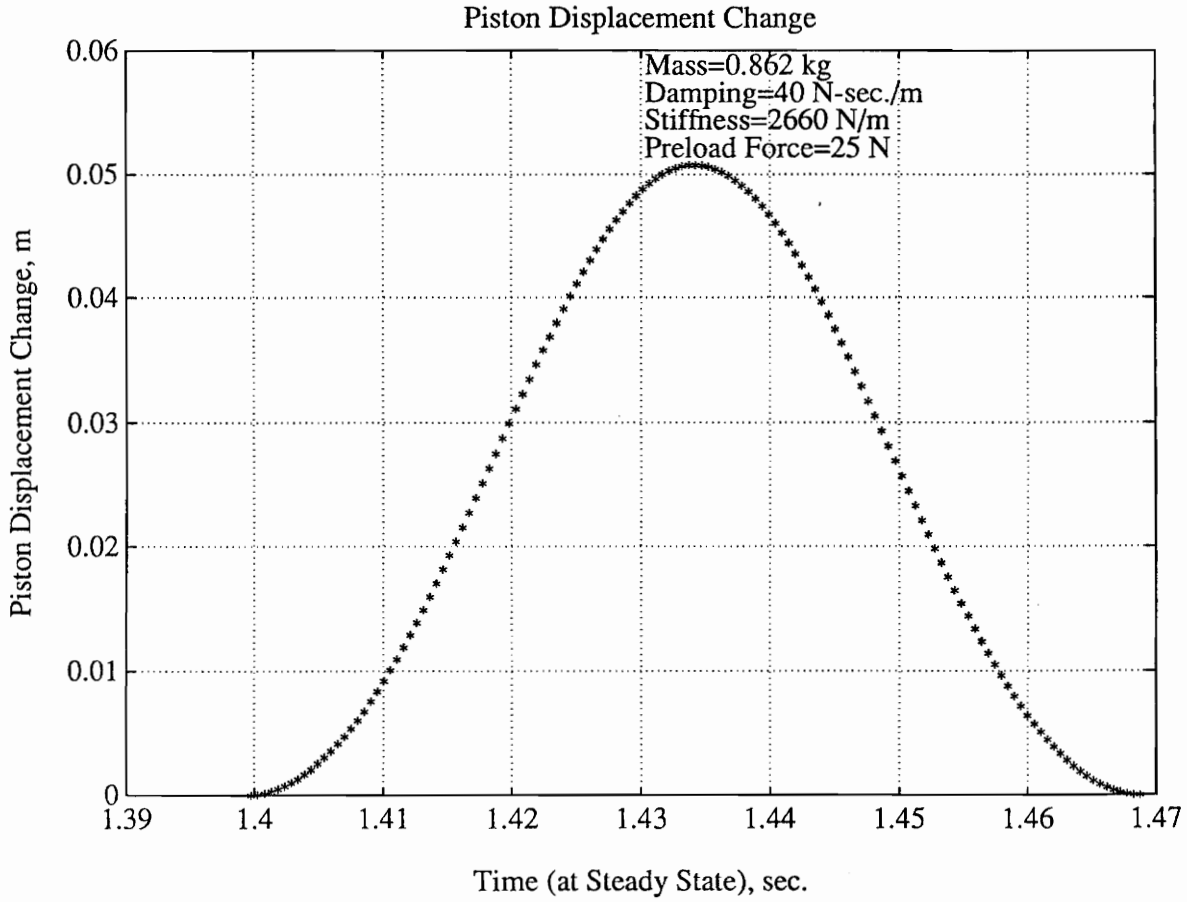


Figure 45. Change of piston displacement

since the mass load torque contains the mass parameter only, at least three independent equations are necessary. As shown in Figures 47, 49, and 51, the low frequency harmonics dominate the equation of motion, accordingly, these equations are set up according to the sine and cosine components of these low frequency harmonics spectra of the motion equation.

6.2.2 Solution for Moment of Inertia and Rotational Damping

The motion equation of the rotor of the motor and reciprocating mechanism is

$$T_{net} - T_{load} = J_s\ddot{\theta} + C_s\dot{\theta} \tag{6.24}$$

where

T_{net} = motor net output torque (N-m); it is the input power minus losses, divided by the angular velocity.

T_{load} = reciprocating mechanism load torque (N-m)

$\ddot{\theta}$ = rotor angular acceleration (rad./ $sec.^2$)

$\dot{\theta}$ = rotor angular velocity (rad./ $sec.^2$)

J_s = total moment of inertia (kg- m^2)

C_s = rotational damping (N-m-sec./rad.)

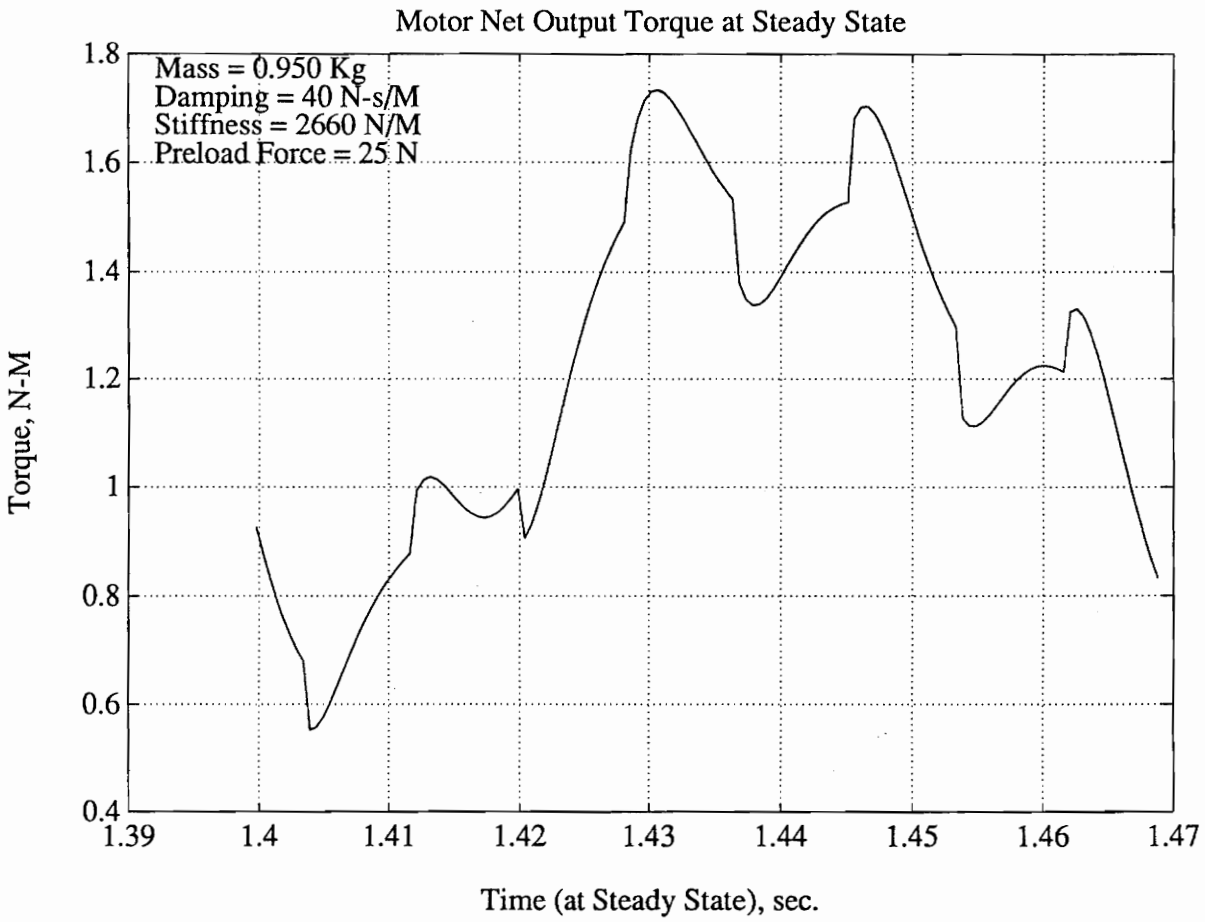


Figure 46. Simulated motor net output torque

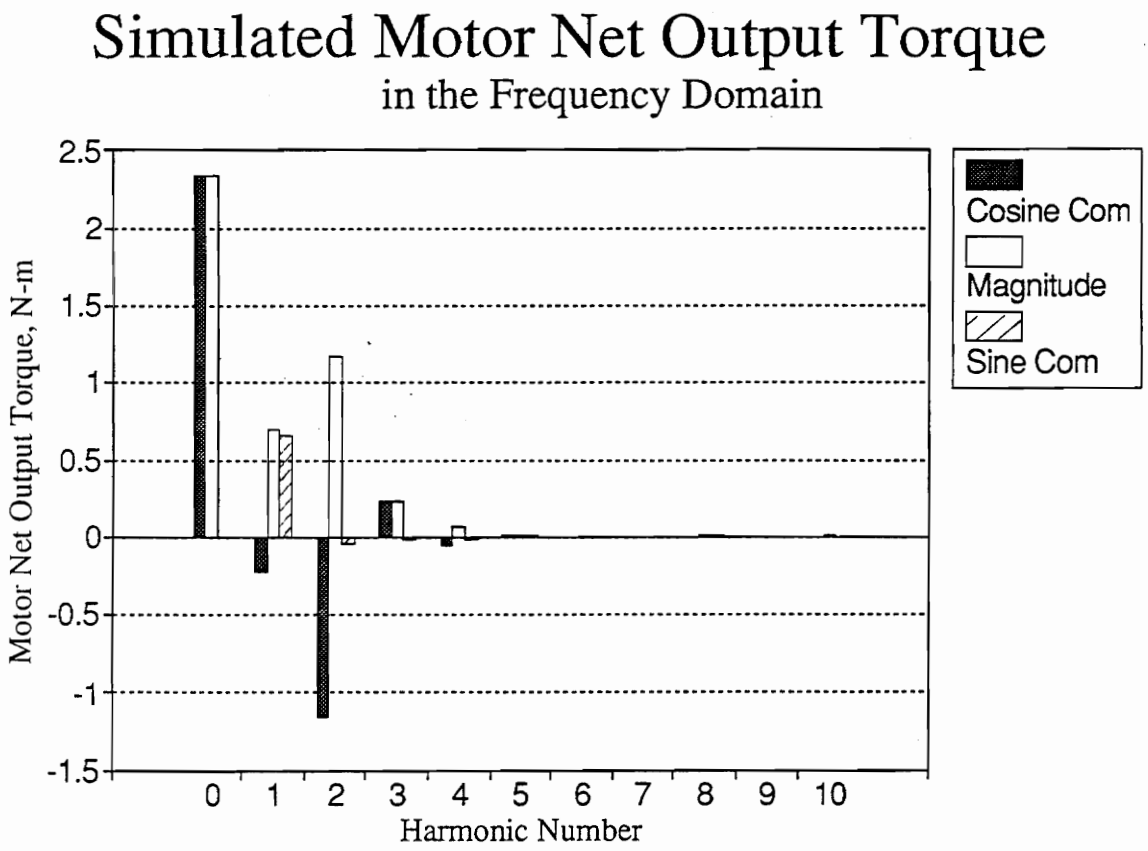


Figure 47. Simulated motor net output torque in frequency domain

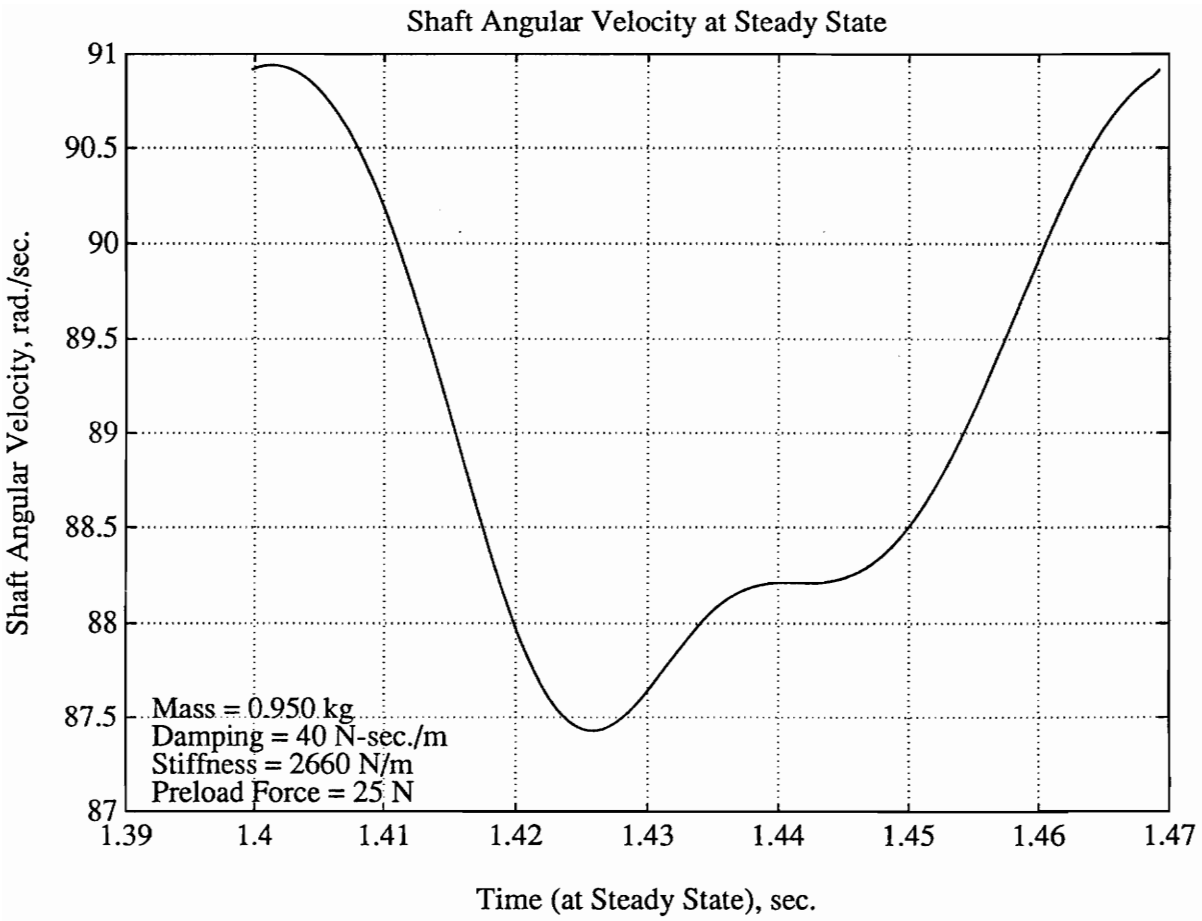


Figure 48. Simulated shaft angular velocity

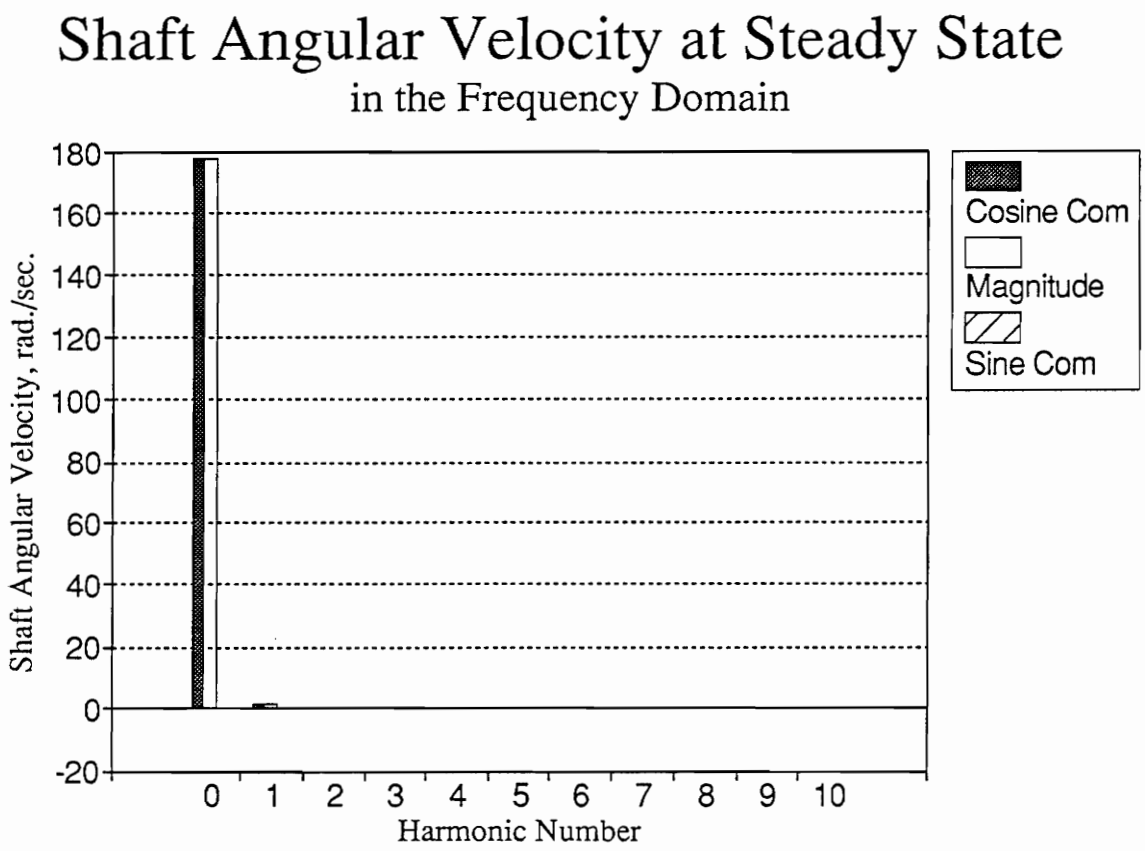


Figure 49. Simulated shaft angular velocity in frequency domain

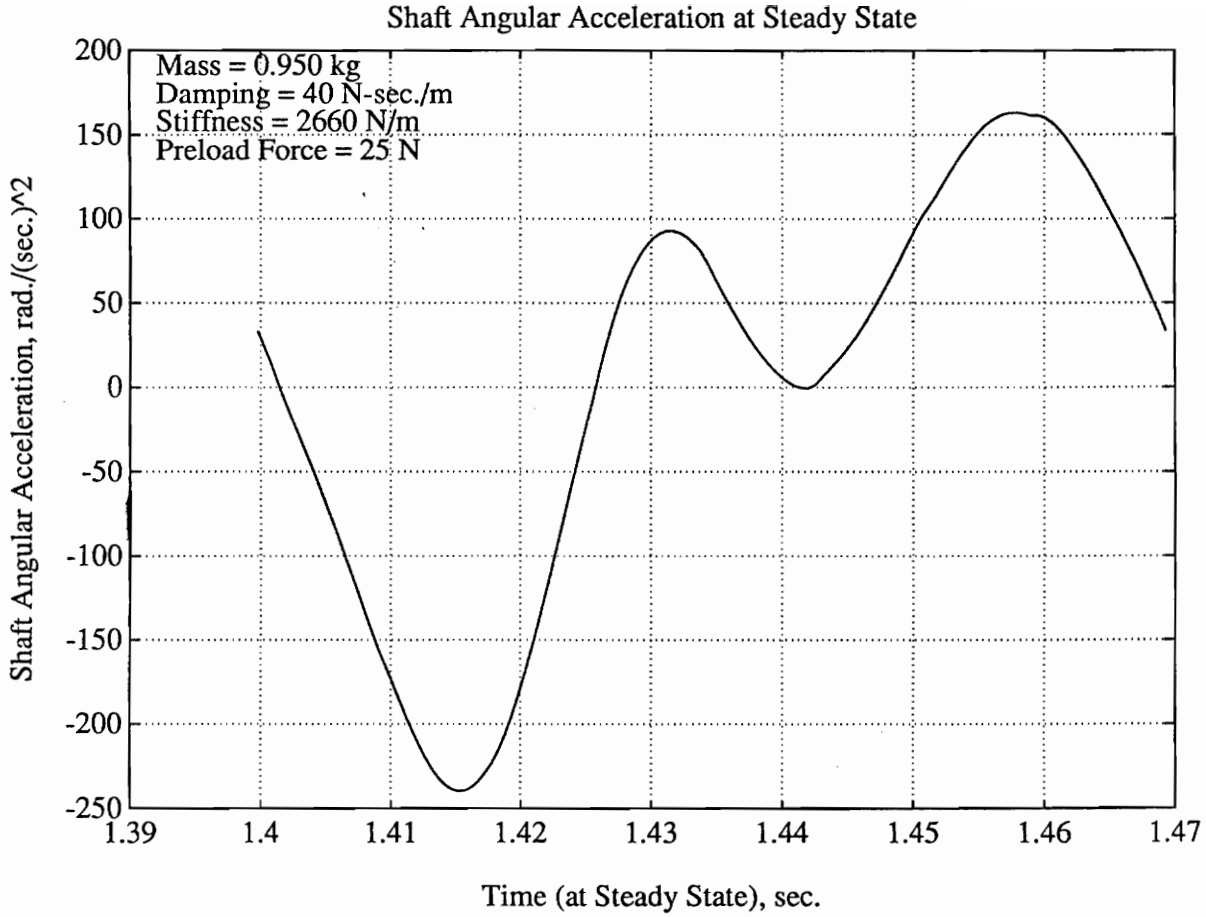


Figure 50. Simulated shaft angular acceleration

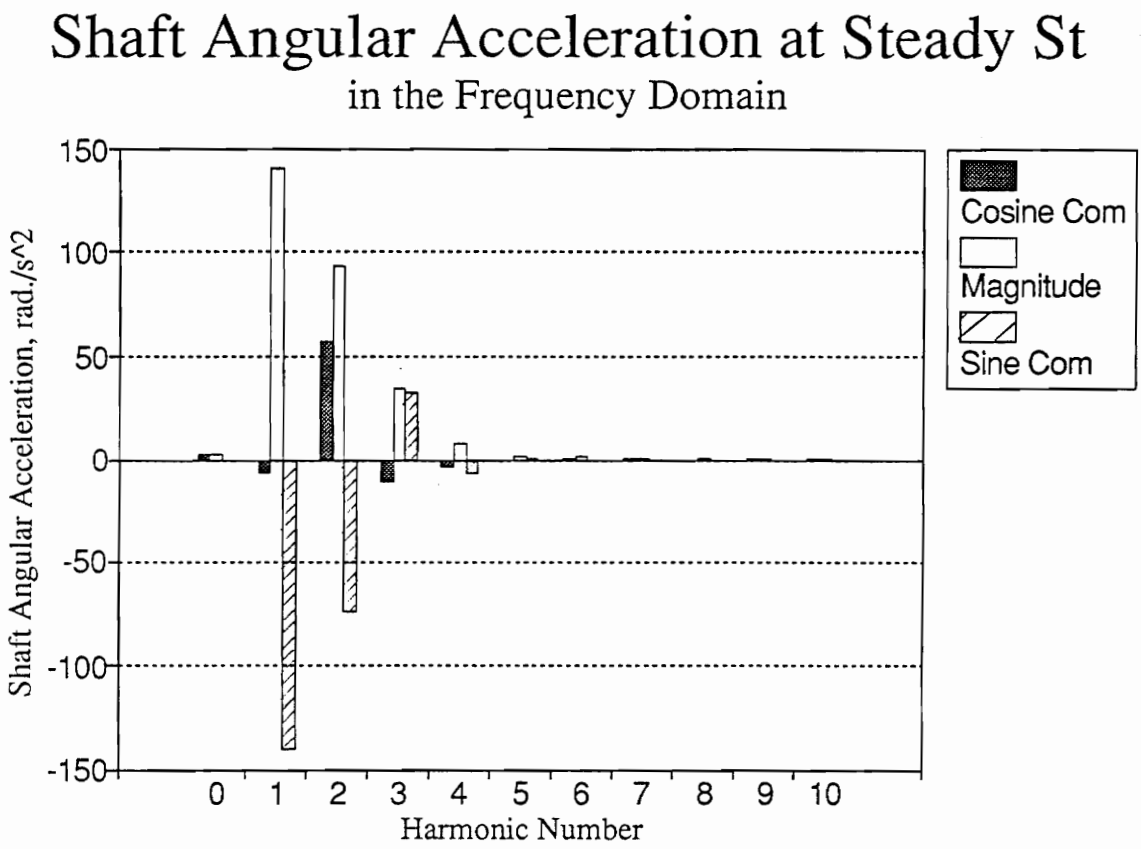


Figure 51. **Simulated shaft angular acceleration in frequency domain**

J_s and C_s are constants; the other terms are time-related variables. Through a Fourier transform, the motion equation can be expressed as

$$\begin{bmatrix} T_{netI} \\ T_{netII} \\ T_{netIII} \\ T_{netIV} \\ T_{netV} \end{bmatrix} = \begin{bmatrix} T_{lmI} & \ddot{\theta}_I & \dot{\theta}_I \\ T_{lmII} & \ddot{\theta}_{II} & \dot{\theta}_{II} \\ T_{lmIII} & \ddot{\theta}_{III} & \dot{\theta}_{III} \\ T_{lmIV} & \ddot{\theta}_{IV} & \dot{\theta}_{IV} \\ T_{lmV} & \ddot{\theta}_V & \dot{\theta}_V \end{bmatrix} \begin{bmatrix} M \\ J_s \\ C_s \end{bmatrix} \tag{6.25}$$

where I, II, III, IV, V (The notation is changed because the excitation force is not the same as that in the example in section 6.1.3.) indicate the sine or cosine components of the specific dominating spectrum in the frequency domain. By substituting the corresponding cosine or sine values in Figures 47, 49, and 51 into the above equation set (Equation 6.25), the parameters J_s, C_s can be accurately solved; (Table 3.) The calculation of the mass will be found in the following section.

6.2.3 Solution for Mechanism Parameters

6.2.3.1 Load Torque Separation

The load torque of the mechanism is not only a function of M, C, K, and F_i, but also a function of displacement X, velocity $\dot{X}$, and acceleration $\ddot{X}$ of the reciprocating mass. The signals of the load torque are divided into components according to the contributions of M, C, K, and F_i .

The load torque can be expressed as

$$T_{load} = T_{lm}M + T_{lc}C + T_{lk}K + T_{lf}F_i \tag{6.26}$$

Table 3. Results of calculated moment of inertia and rotational damping

Moment of inertia J_s (kg - m^2)			Rotational damping C_s (N-m-sec.)		
True value	Calculated value	Error %	True value	Calculated value	Error %
0.02000	0.02018	0.9	0.01000	0.00986	-1.4
0.02000	0.01940	-3.0	0.01000	0.00998	-0.2
0.02000	0.02003	0.15	0.01000	0.00978	-2.2

and the components of mass M, damping C, spring stiffness K, and preload force F_i are

$$T_{lm} = A(\ddot{X} + g\mu) \tag{6.27}$$

$$T_{lc} = A\dot{X} \tag{6.28}$$

$$T_{lk} = A(X - (L - R)) \tag{6.29}$$

$$T_{lf} = A \tag{6.30}$$

where

$$A = \frac{R(\sin\theta - \cos\theta \tan\beta)}{(1 - \mu \tan\beta)}$$

μ = dynamic friction coefficient between slider mass and slideway.

g = 9.81 m/ $sec.^2$

Then the previous motor-mechanism motion equation (Equation 6.24) becomes

$$T_{net} - J_s\ddot{\theta} - C_s\dot{\theta} = T_{lm}M + T_{lc}C + T_{lk}K + T_{lf}F_l \tag{6.31}$$

The frequency domain spectra of T_{lm}, T_{lc}, T_{lk} and T_{lf} are shown in the following figures.

6.2.3.2 Motion Equation in the Frequency Domain

The time-domain motion equation

$$T_{net} - J_s\ddot{\theta} - C_s\dot{\theta} = T_{lm}M + T_{lc}C + T_{lk}K + T_{lf}F_i$$

is performed with a DFT; the spectra of the dominating sine and cosine components (as shown in the figures) can be used to calculate the unknown M, C, K, F_i :

$$T_{netI} - J_s\ddot{\theta}_I - C_s\dot{\theta}_I = T_{lmI}M + T_{lcI}C + T_{lkI}K + T_{lfI}F_i \tag{6.32}$$

$$T_{netII} - J_s\ddot{\theta}_{II} - C_s\dot{\theta}_{II} = T_{lmII}M + T_{lcII}C + T_{lkII}K + T_{lfII}F_i \tag{6.33}$$

$$T_{netIII} - J_s\ddot{\theta}_{III} - C_s\dot{\theta}_{III} = T_{lmIII}M + T_{lcIII}C + T_{lkIII}K + T_{lfIII}F_i \tag{6.34}$$

$$T_{netIV} - J_s\ddot{\theta}_{IV} - C_s\dot{\theta}_{IV} = T_{lmIV}M + T_{lcIV}C + T_{lkIV}K + T_{lfIV}F_i \tag{6.35}$$

$$T_{netV} - J_s\ddot{\theta}_V - C_s\dot{\theta}_V = T_{lmV}M + T_{lcV}C + T_{lkV}K + T_{lfV}F_i \tag{6.36}$$

Therefore, the above equations (Equations 6.32 through 6.36) can be expressed as

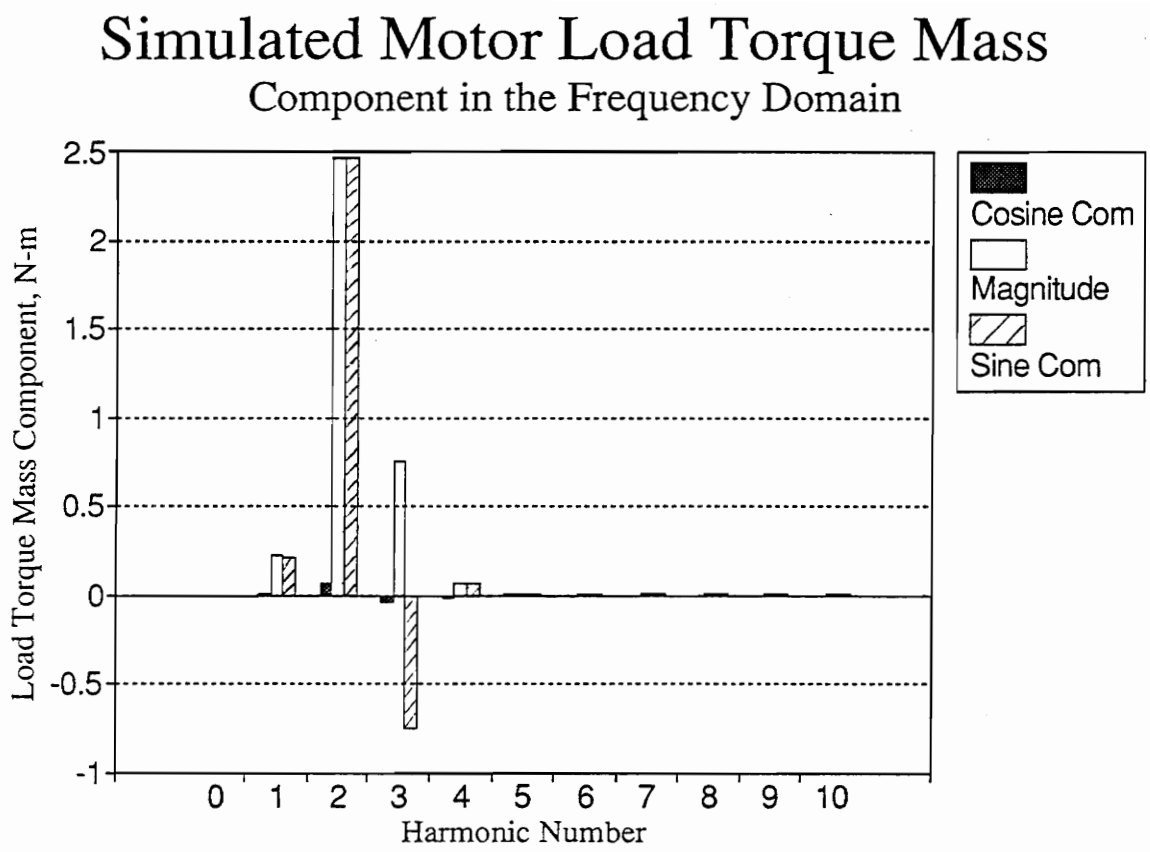


Figure 52. Simulated motor load torque mass component in the frequency domain

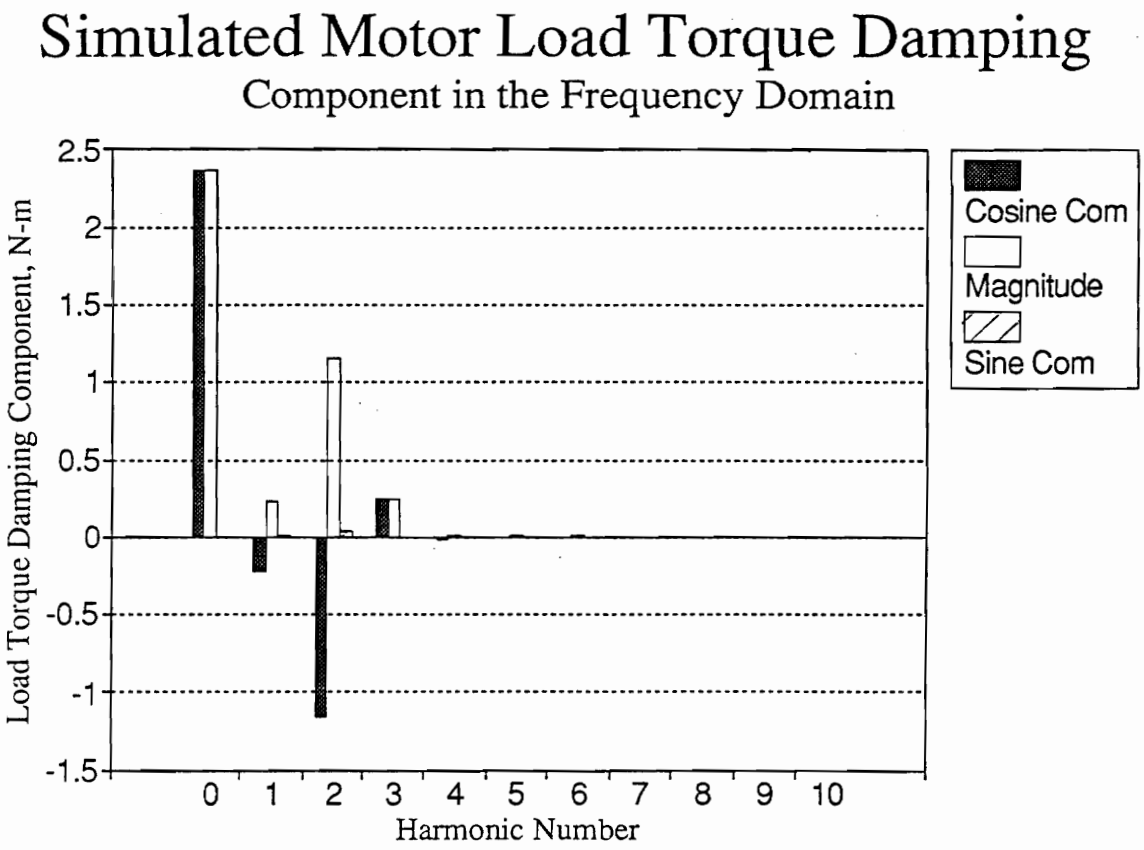


Figure 53. Simulated motor load torque damping component in the frequency domain

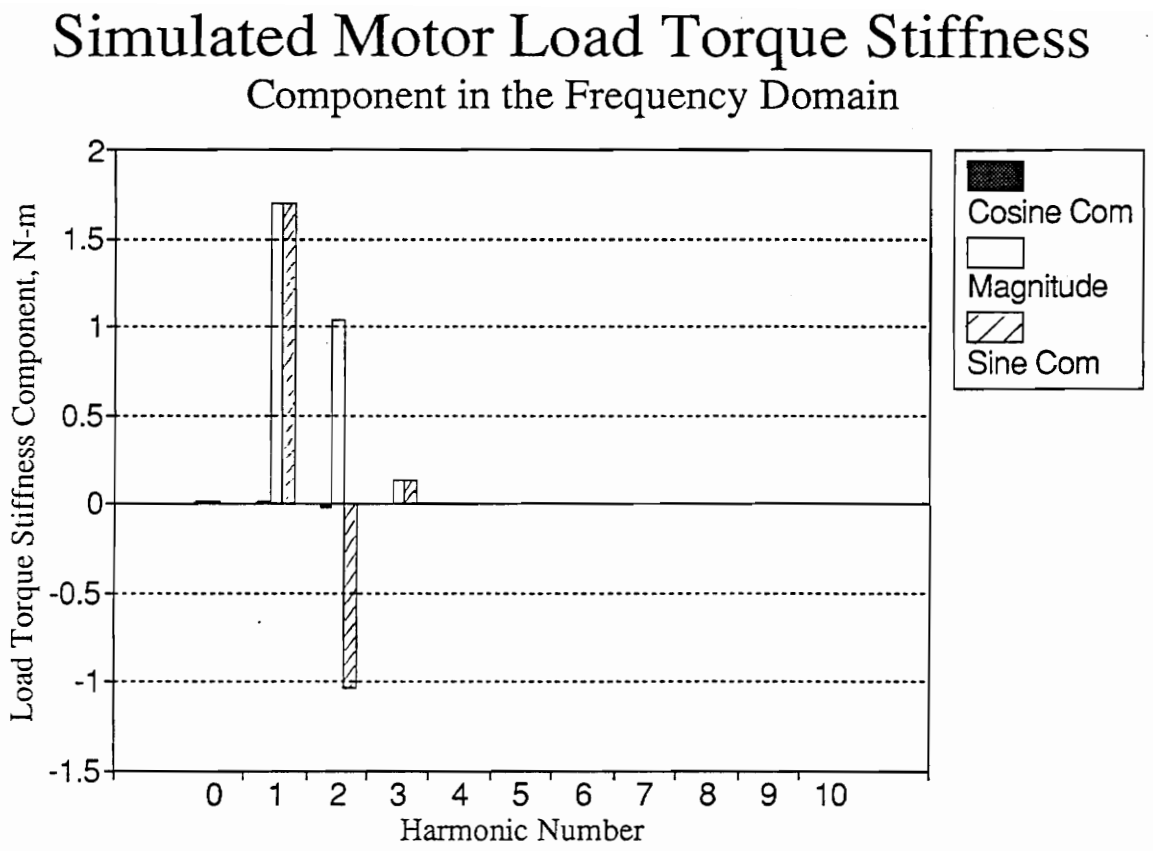


Figure 54. Simulated motor load torque stiffness component in the frequency domain

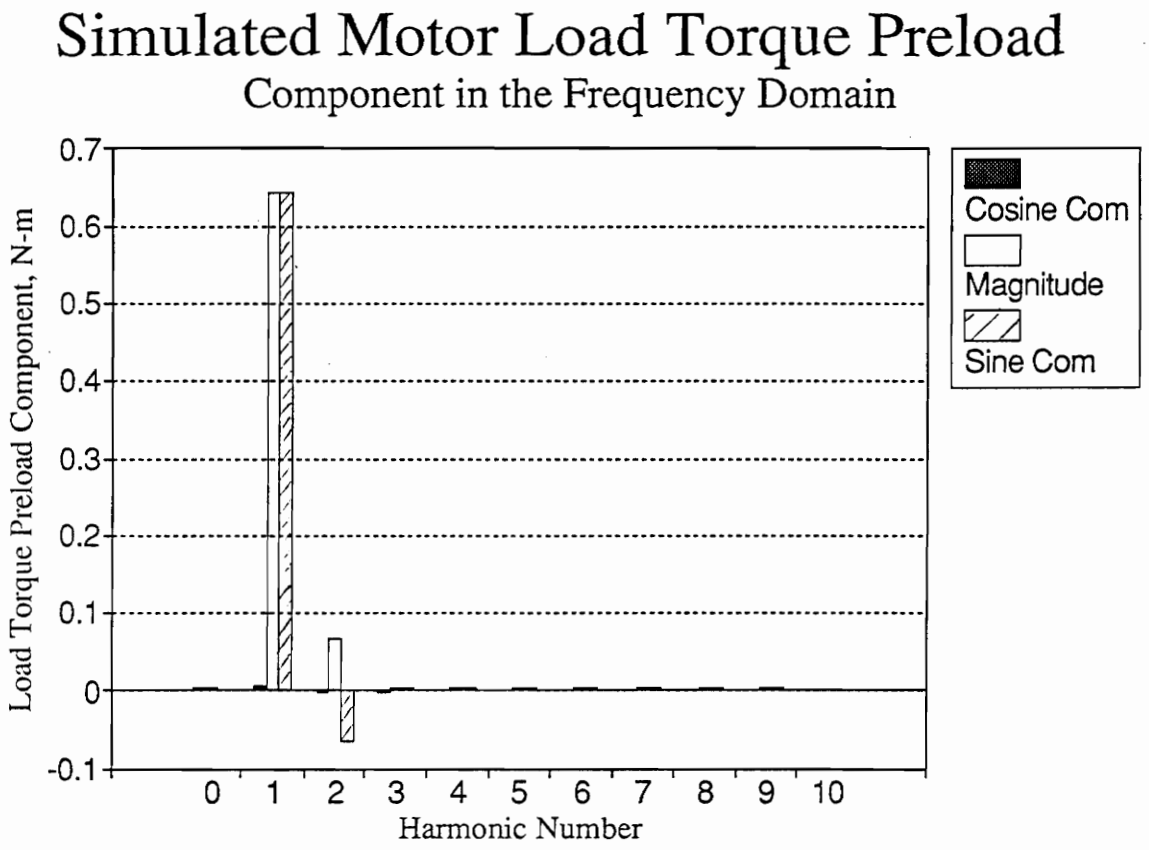


Figure 55. Simulated motor load torque preload force component in the frequency domain

$$\begin{bmatrix} T_{netI'} \\ T_{netII'} \\ T_{netIII'} \\ T_{netIV'} \\ T_{netV'} \end{bmatrix} = \begin{bmatrix} T_{lmI} & T_{lcI} & T_{lkI} & T_{lfI} \\ T_{lmII} & T_{lcII} & T_{lkII} & T_{lfII} \\ T_{lmIII} & T_{lcIII} & T_{lkIII} & T_{lfIII} \\ T_{lmIV} & T_{lcIV} & T_{lkIV} & T_{lfIV} \\ T_{lmV} & T_{lcV} & T_{lkV} & T_{lfV} \end{bmatrix} \begin{bmatrix} M \\ C \\ K \\ F_i \end{bmatrix} \tag{6.37}$$

where

$$T_{net'} = T_{net} - J_s\ddot{\theta} - C_s\dot{\theta} \tag{6.38}$$

and I, II, III, IV, V indicate the sine and cosine components of the dominating harmonics of the transformed frequency domain spectra. Because the number of the equations exceeds the number of the unknowns, a least squares algorithm is used to solve for the unknowns.

The objective of the least squares algorithm program is to solve the unknown parameters M, C, K, and F_i in the motion equation. Recall that the transformed motion equation: the matrices of T_{lm}, T_{lc}, T_{lk}, T_{lf} and of T_{net} are regarded as known values. The unknown parameters M, C, K, and F_i can be accurately solved; (Table 4, and Table 5)

6.2.3.3 Precision Discussion

Relationship of the Magnitude and the Relative Error

The results of the determination of the damping constant and preload force are shown in Figure 57 and Figure 58. The relative errors are high when the values to be calculated are small. The errors may be attributed to the smaller load torque contributions from these two parameters. Recall the load torque equation

Table 4. Results of calculated lumped mass

Lumped mass (kg)		
True value	Calculated value	Error %
0.862	0.8649	0.34
0.868	0.8700	0.23
0.875	0.8749	-0.01
0.880	0.8801	0.01
0.900	0.8987	-0.14
0.920	0.9185	-0.16
0.925	0.9273	0.25
0.940	0.9387	-0.14
0.945	0.9475	0.26
0.950	0.9485	-0.16
0.960	0.9587	-0.14
0.986	0.9877	0.17
0.995	0.9990	0.40
1.862	1.8658	0.20

$$T_{load} = T_{lm}M + T_{lc}C + T_{lk}K + T_{lf}F_i \tag{6.39}$$

where

M = lumped mass of the piston and the connecting rod

C = damping constant

K = spring stiffness

F_i = preload force

Table 5. Results of calculated spring stiffness

Spring stiffness (N/m)		
True value	Calculated value	Error %
260	279	7.31
2600	2632	1.23
2660	2675	0.56
2730	2717	-0.48
2760	2749	-0.40
2770	2761	-0.32
2850	2841	-0.32
2880	2871	-0.31

In the above case, since the contribution of $T_{lm}M$ and $T_{lk}K$ are about one order of magnitude larger than the other two terms, the calculated values of mass and spring stiffness are almost the same as the true values.

The results show that the calculated parameters of the damping constant and the preload force are more accurate as their magnitudes increase.

Effect of the Load Torque Contribution

Figure 59 and Figure 60 show that the components' contributions to the load torque will influence the accuracies of the calculated parameters. Figure 59 shows that if the preload force is 4 N, then its contribution is rather small. The calculated result is 70 percent of the true value. The same situation happens in the damping calculation. If the damping is 4 N-sec./m, then its contribution is also small, and the calculated value is 87 percent of the true value.

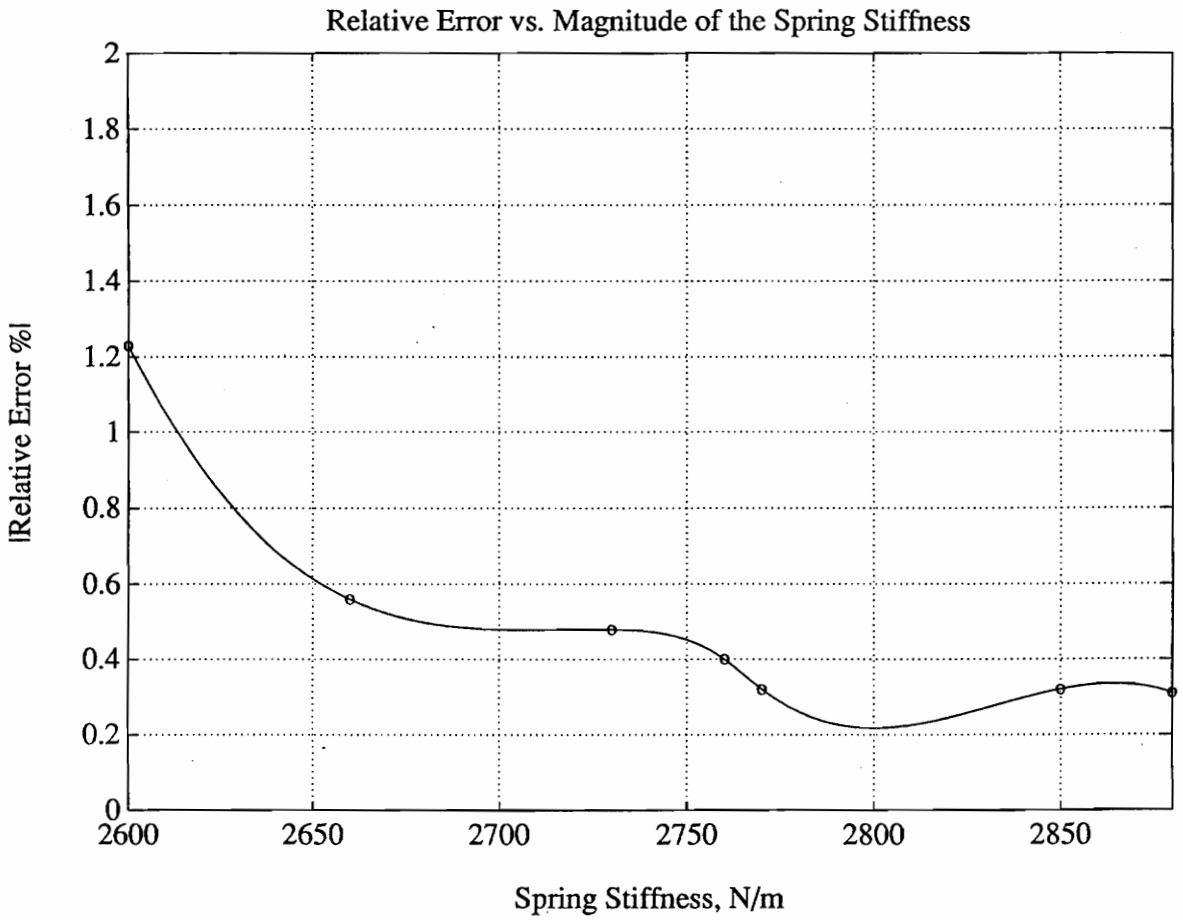


Figure 56. Relative error % vs. magnitude of spring stiffness (N/m)

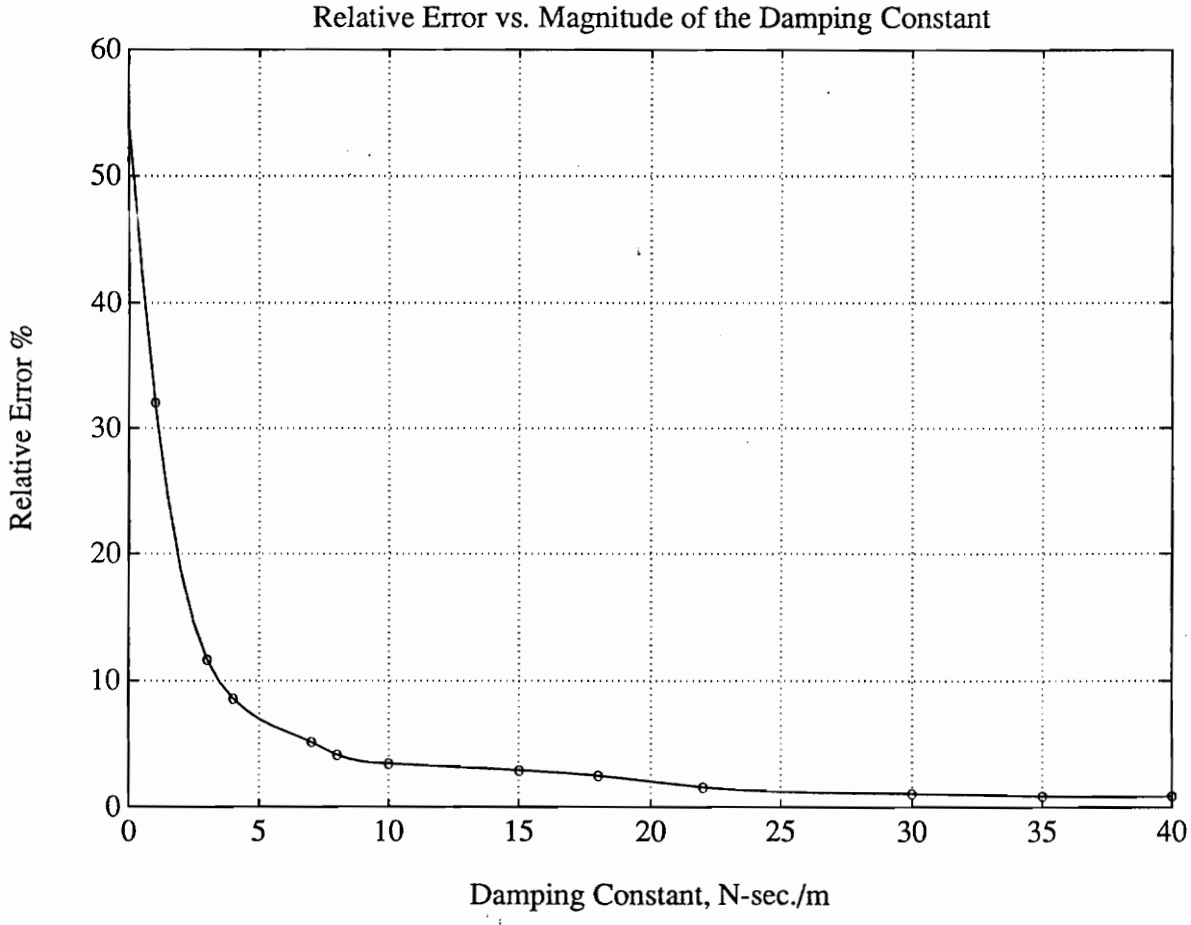


Figure 57. **Relative error % vs. magnitude of damping constant (N-sec./m)**

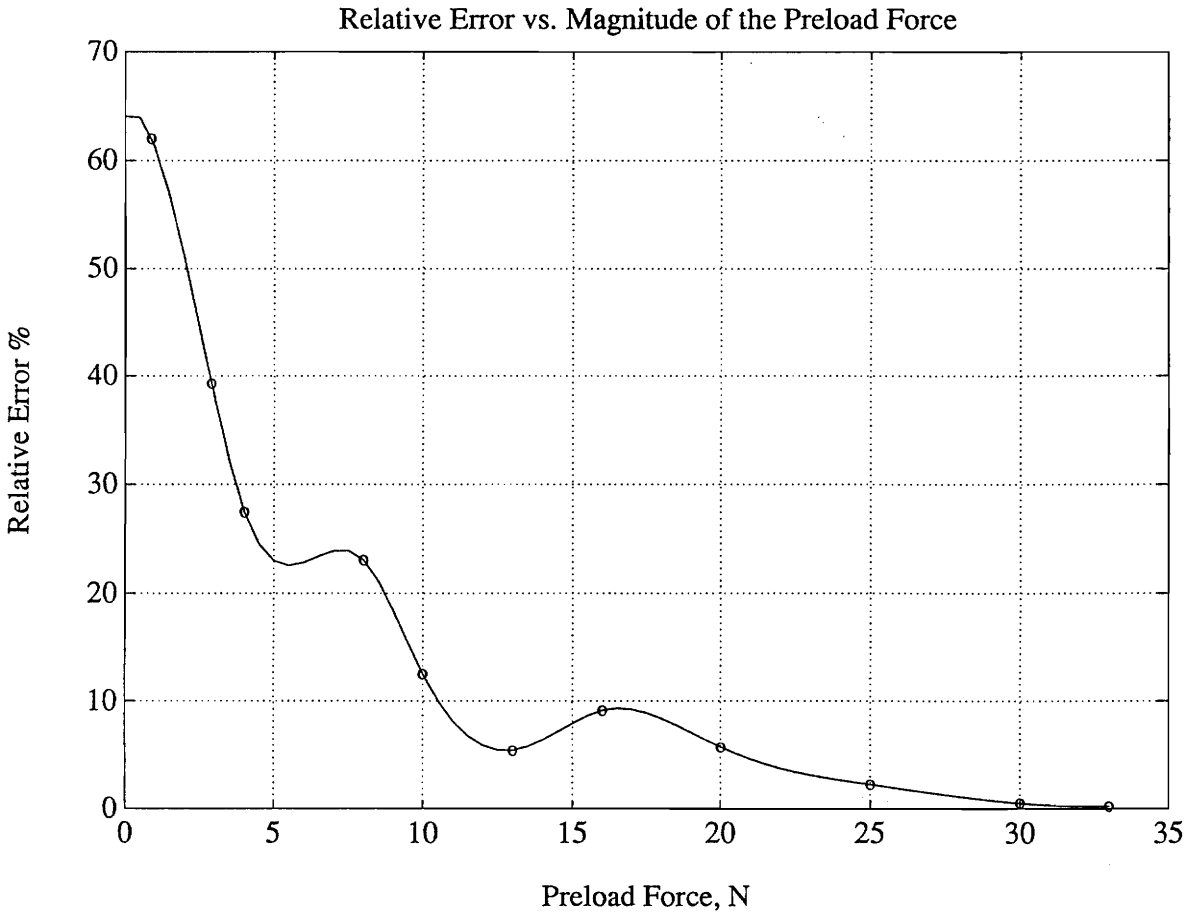


Figure 58. Relative error % vs. magnitude of preload force (N)

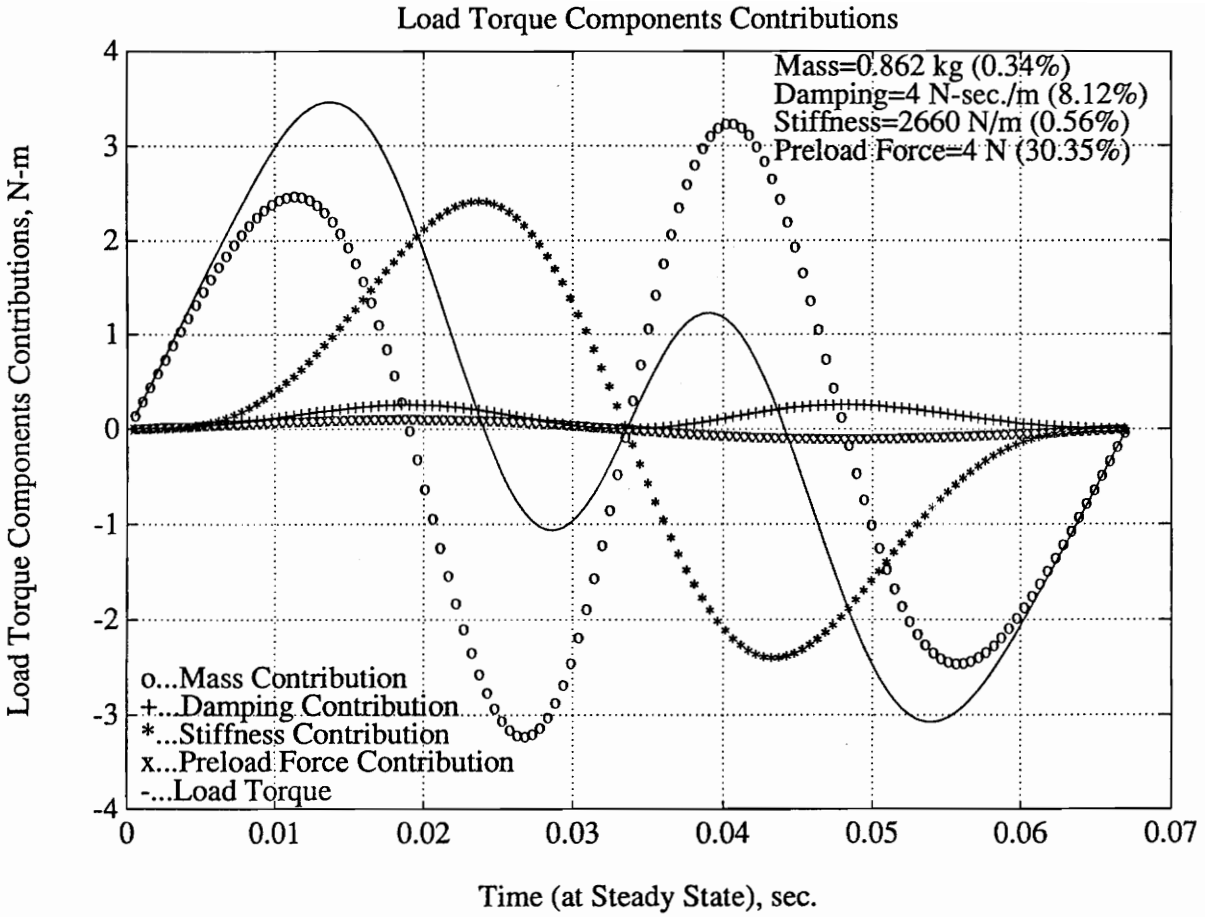


Figure 59. Torque contribution plot of the motor reciprocating-mechanism

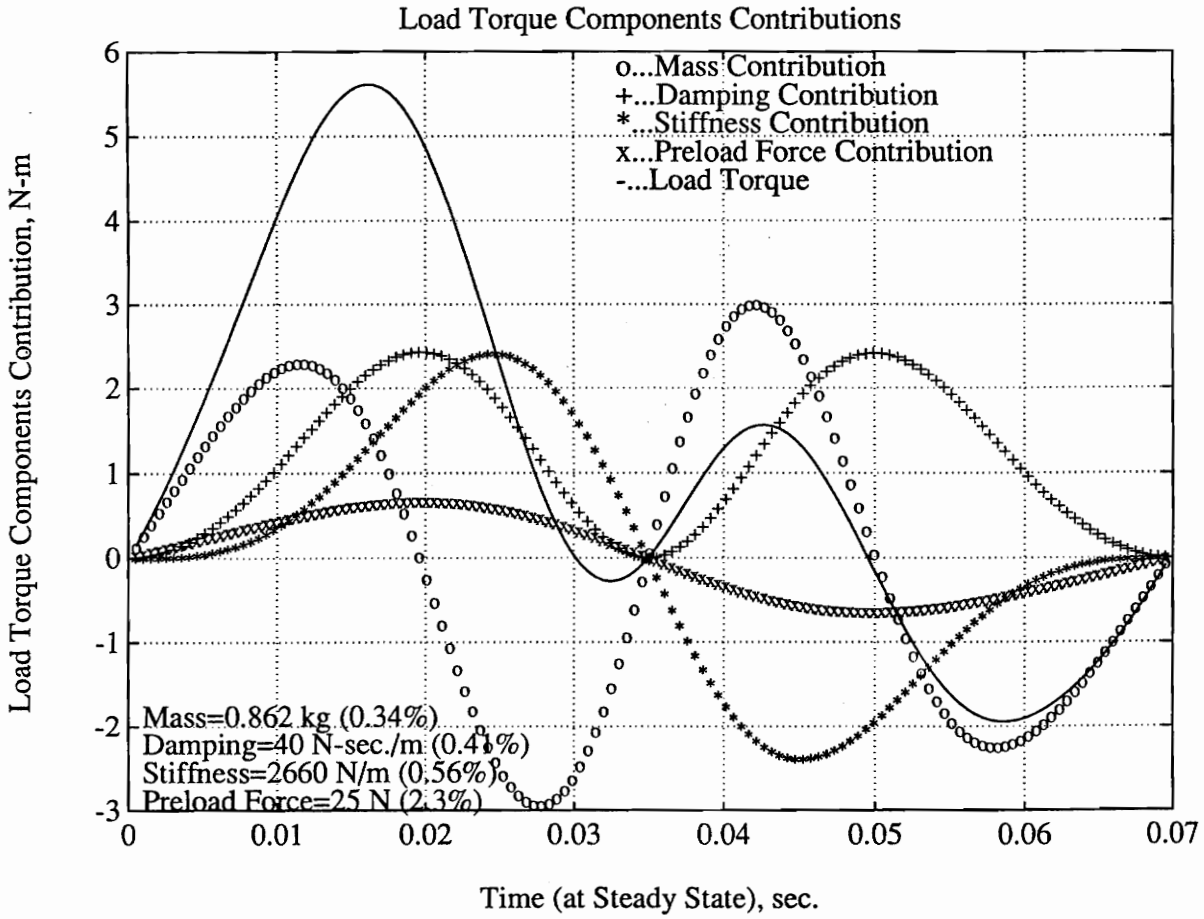


Figure 60. Torque contribution plot of the motor reciprocating-mechanism

Figure 60 shows that when the contributions of the damping and preload force are larger, their calculated values are greatly improved to 99.1 percent and 97.7 percent, respectively.

Therefore, according to Figures 59 and 60, it is sufficient to say that the load torque contribution of a parameter will influence the accuracy of the calculated value of the parameter.

Chapter 7

Conclusions and Recommendations

7.1.1 Conclusions

This research described the modeling of an induction motor to account for losses and identified the reciprocating mechanism parameters. The conclusions drawn from this research are :

1. The mechanism load torque derivation showed that the load torque can be divided into the components contributed by mass, damping, spring stiffness, and preload force.

2. The motor core loss, the sum of the hysteresis and eddy current losses, can be simulated as a resistance load in the motor differential equations.

3. The modified Stanley motor differential equations proved to be effective.

4. Since a reciprocating machine operates periodically, we can take advantage of sampling multiple-period waveforms to avoid leakage problems.

5. The results showed that the calculated mass and stiffness can be accurately determined within 1% (mass larger than 0.85 kg, spring stiffness larger than 2620 N/m) of the true value. Damping and preload force can be determined within 5% (C larger than 7 n-sec./m) and 10% (F_i larger than 11 N), respectively (Figures 56, 57, and 58). The accuracy of the calculated damping constant (or preload force) has been found to be related to the damping constant (or preload force) load torque contribution. It is certain that the larger the parameter (C, F_i) load torque contribution, the more accurate is the calculated parameter. (as shown in Figures 57 and 58)

6. The results showed that the calculated total rotary moment of inertia (J_s) could be accurately solved to within 3% and the rotational damping (C_s) within 2% of the true value without fail.

7. The results showed that the load torque of the mechanism is an arithmetic sum of the contribution of mass, damping, spring stiffness, and preload force (Equation 6.26).

7.1.2 Recommendations

The identification of the parameters could lead to advancements in reciprocating machine diagnosis in the future. The solution technique documented in this thesis may provide a way to access the internal condition of the machine.

Since this research is based entirely on a simulated reciprocating mechanism, further efforts must be made to apply the methods presented herein to real operating machines (compressor, engine, etc.)

The researcher should:

1. Experimentally investigate all forces on the machine to model the load torque of the machine. These forces will the be mass, stiffness, damping, and preload forces. For example, the pistons result in the mass forces in a machine. At the same time, the piston ring produces a preload force in the system.

2. Determine the relationship between the variation of the parameters and the machine failure. The influence of parameter change on machine failure could be various. Having the same percentage change of each parameter would result in different operating conditions.

Bibliography

1. Sears, F.W., Zemansky, M.W. and Young, H.D., *University Physics*, 7th Ed. Addison-Wesley Pub. Co., 1987

2. Cochran, Paul L., *Polyphase Induction Motors: Analysis, Design, and Application*, M. Dekker, 1989

3. Del Toro, Vincent, *Basic Electric Machines*, Prentice Hall, 1990

4. Chapman, S.J., *Electric Machinery Fundamentals*, McGraw-Hill, 1985

5. Nasar, S.A. and Unnewehr, L.E., *Electro-Mechanics and Electric Machines*, 2nd Ed. John Wiley & Sons., 1983

6. Honeycutt, R.A., *Electromechanical Devices*, Prentice Hall, 1983

7. Fitzgerald, A.E., et al, *Electric Machinery*, 4th Ed. McGraw-Hill, 1983

8. Stein, R. and Hunt, William T. Jr., *Electric Power System Components*, Van Nostrand Reinhold Co., 1979

9. Bourne, R, "No load method of estimating stray load loss in small cage induction motors," *IEE PROCEEDINGS*, V.136, pt. B, No.2, March, 1989, pp.92-95

10. Stanley, H.C., "An Analysis of the Induction Machine," *AIEE Transactions*, V. 57, 1938, 751-757

11. Bendat, Julius S. and Piersol, Allan G., *Random Data*, 2nd Ed. John Wiley & Sons, Inc. 1986

12. Brigham, E. Oran, *The Fast Fourier Transform and its Application*, Avantek Inc., 1988

13. Dally, J. W. et al, *Instrumentation for Engineering Measurements*, John Wiley & Sons, 1984

14. Williams, Jeremy, *Parameter Identification in a Slider-Crank Mechanism Through Direct Measurement of Shaft Velocity*, Masters Thesis, VPI&SU, 1989

15. Cheney, Ward and Kincaid, David, *Numerical Mathematics and Computing*, 2nd Ed. Brooks/Cole Publishing Co., 1985

16. IMSL MATH/LIBRARY, *FORTRAN Subroutines for Mathematical Application*, User's Manual Version 1.1, December 1989

17. Kanth, R. M., *A Personal Computer Based Instrumentation System for Determining Real-Time Dynamic Torque in Rotating Machinery*, Masters Thesis, VPI&SU, 1987

18. Lipo, T.A.,"Modeling and Simulation of Induction Motors with Saturable Leakage Reactances," *IEE Trans. on Industry Applications* , V. IA-20, No.1, January/February 1984, pp.180-189

19. Thomson, T. William, *Theory of Vibration with Applications*, 2nd Ed. Prentice-Hall, Inc., 1981

20. Åström, K. J. and Eykhoff, P., "System Identification-A Survey," *Automatica* , V. 7, 1971, pp.123-162

21. Lim, T. G. et al, "Parameter Identification Method for Robot Dynamic Models Using a Balancing Mechanism," *Robotica* , V. 7, Part 4, 1989, pp.327-337

22. Driels, M. R. et al, "Significance of Observation Strategy on the Design of Robot Calibration Experiments," *Journal of Robotic Systems* , V. 7, No. 2, 1990, pp.197-223

Appendix A
Program Listing

A.1 Execution file of Simulation Program for IBM/VM

```
/* MTRE EXEC TO RUN PLOT 10 PROGRAM ON Y-TERM TEK EMULATOR */
 'GLOBAL TXTLIB VSF2FORT PREVIEW TCSLIB'
 'GLOBAL LOADLIB VSF2LOAD'
/* COMPILES MOTOR SIMULATION PROGRAM; MTR FORTRAN */
/*'FORTVS2 MTR'*/
/*IF RC ¬ = 0 THEN EXIT RC*/
/* DEFINE INPUT AND OUTPUT DISKS */
 ' FILEDEF 9 DISK TEST MTR A'
 ' FILEDEF 10 DISK TELTLM DAT G'
 ' FILEDEF 11 DISK THOMALPH DAT G'
/* FOR M, C, K, F SOLVING */
 ' FILEDEF 12 DISK TLMCKF DAT G'
 ' FILEDEF 13 DISK TOUTJA DAT G'
 ' FILEDEF 14 DISK DVA DAT G'
 ' FILEDEF 15 DISK COETAN DAT G'
 ' FILEDEF 16 DISK ANNETLOD DAT G'
 'LOAD MTR (NOMAP'
 'START'
```

A.2 Motor and Reciprocating Mechanism Simulation Program

```
C
C    PROGRAM FOR THE DYNAMIC ELECTRICAL RESPONSE OF A
C    TWO AXIS ELECTRIC MOTOR MODEL
     REAL I,J,JS,KS,NPOLES,Y(6),DY(6),XMUL,IR,M,K
     COMMON/ELEC/I(3),V(3),DELT,TELEC,TLOAD,IR(3),RH
C
C    COMMON FOR PLOTTING DATA
C
```

```
      COMMON/PLOTT/X(963,17),DYSAV(6)
C
C     MOTOR CONSTANTS
C
      COMMON/MOTOR/X1,X2,R1,R2,XM,J,FREQ,DFREQ,NPOLES,VRMS,FLT,JS,KS,CS
     +     ,M,C,K,FORCE,CDMPNG
C
C     OPEN MOTOR DATA FILE FOR UNFORMATTED READ
C
C     ORDER OF DATA: X1, X2, R1, R2, XM, J, FREQ, DFREQ, NPOLES, VRMS,
C                    FLT, JS, KS, CS
C
C
      DATA TWOPI/6.2831853/
      READ(9,*,END=999) X1,X2,R1,R2,XM,J,FREQ,DFREQ,NPOLES,VRMS,FLT,JS,
     1     KS,CS,M,C,K,FORCE,CDMPNG,STIME,ENTIME
C
C     INTEGRATION TIMER VARIABLES
C
C     STIME - START TIME FOR SIMULATION
C     ENTIME - END TIME FOR SIMULATION
C     TIME   - TIME VARIABLE, UPDATED BY RKG1
C     DELT   - TIME STEP
C
99    CONTINUE
      IF(ENTIME.EQ.0.) GO TO 999
      DELT=(1./(5.*FREQ))/(4.*NPOLES)
C
C     INITIALIZE MOTOR ELECTRICALS
C
      NN=0
      CALL DRV0
      NEQ=6
      TIME=0.
      DO 2 LP=1,6
      Y(LP)=0.
2     DY(LP)=0.
C
C     DETERMINE COUNTER FOR OUTPUT
C
      NCNT=1
5     NPLPTS=(ENTIME-STIME)/(FLOAT(NCNT)*DELT)
      IF(NPLPTS.LT.1000) GO TO 4
      NCNT=NCNT+1
      GO TO 5
4     CONTINUE
C
C     INITIALIZE GRAPHICS
C
      DELANGL=DELT/ENTIME*TWOPI
      ANGLE=TWOPI/4.+DELANGL
      KOUT=0
      NOUT=0
C
C     BEGIN LOOP
C
```

```
C     DYNAMIC SECTION
C
1     CONTINUE
      ANGLE = ANGLE-DELANGL
      IF(NOUT.NE.0) GO TO 6
      KOUT = KOUT + 1
      IF(KOUT.NE.10) GO TO 6
      KOUT = 0
      IXX = INT(3.*COS(ANGLE))
      IYY = INT(3.*SIN(ANGLE))
      JXX = 3 + IXX
      JYY = 4 + IYY
      XMUL = 2.0
      IF(TIME.LT.STIME) XMUL = 1.8
      KXX = 3 + INT(XMUL*FLOAT(IXX))
      KYY = 4 + INT(XMUL*FLOAT(IYY))
6     NOUT = NOUT + 1
C
C     INTEGRATE INTEGRATE INTEGRATE INTEGRATE
C
      CALL RKG1L(NEQ,TIME,Y,DY)
C
C     IF TIME IS PROPER, OUTPUT DESIRED VARIABLES
C
      IF(NOUT.EQ.1.AND.TIME.GE.STIME) CALL SAVE(NN,TIME,Y,DY)
C
C     TEST TIME TO SEE WHETHER SIMULATION IS COMPLETE
C
      IF(NOUT.EQ.NCNT) NOUT = 0
      IF(TIME.LT.ENTIME) GO TO 1
C     CALL DISPLAY
999   STOP
      END
C
      SUBROUTINE RKG1L(NEQ,TIME,Y,DY)
C
C     RUNGE-KUTTA 4TH ORDER, FIXED STEP SIZE ROUTINE
C     PROVIDED BY R. G. LEONARD, VPI&SU ME DEPT
C
      REAL I,Y(6),DY(6),IR
      DIMENSION A(2),Q(100)
      COMMON/ELEC/I(3),V(3),H,TELEC,TLOAD,IR(3),RH
C
C     REAL*8 CONSTANTS
C
C     A(1) = 0.292893218813452475
C     A(2) = 1.70710678118654752
C
      A(1) = 0.2928932
      A(2) = 1.7071068
      H2 = 0.5*H
      CALL DERIV(NEQ,TIME,Y,DY)
      DO 13 II = 1,NEQ
      B = H2*DY(II)-Q(II)
      Y(II) = Y(II) + B
13    Q(II) = Q(II) + 3.0*B-H2*DY(II)
```

```
      TIME = TIME + H2
      DO 20 J = 1,2
      CALL DERIV(NEQ,TIME,Y,DY)
      DO 20 II = 1,NEQ
      B = A(J)*(H*DY(II)-Q(II))
      Y(II) = Y(II) + B
   20  Q(II) = Q(II) + 3.0*B-A(J)*H*DY(II)
      TIME = TIME + H2
      CALL DERIV(NEQ,TIME,Y,DY)
      DO 26 II = 1,NEQ
      B = (H*DY(II)-2.0*Q(II))/6.0
      Y(II) = Y(II) + B
   26  Q(II) = Q(II) + 3.0*B-H2*DY(II)
      RETURN
      END
C
      SUBROUTINE DERIV(NEQ,X,Y,DY)
C
C     THIS ROUTINE, GIVEN THE FORMATION FOR THE DERIVATIVE, WILL
C     COMPUTE THE DERIVATIVE VECTOR, DY, ON THE BASIS OF THE Y
C     VECTOR AND OTHER PROVIDED VARIABLES.
C
      REAL J,LDSSTAR,LQSSTAR,IQR,IDR,IQS,IDS,L1,L2,L3,LM,LQM,LDM,
     1   I,NPOLES,N,X,Y(6),DY(6),XTMP,KS,JS,IR,M,K
      COMMON/ELEC/I(3),V(3),H,TELEC,TLOAD,IR(3),RH
      COMMON/INIT/VOLTS,OMEGAS,OMEGAD,N,VQS,VDS,IQS,IDS,IDR,IQR,
     1  L1,RS,RR,L2,LM
      COMMON/MOTOR/X1,X2,R1,R2,XM,J,FREQ,DFREQ,NPOLES,VRMS,FLT,JS,KS,CS
     +     ,M,C,K,FORCE,CDMPNG
      COMMON/PLOTT/XTMP(963,17),DYSAV(6)
      DATA TWOPI/6.2831853/
C
C     GET MECHANICAL LOAD
C
      CALL LOAD(TLOAD,Y(6),Y(5),DY(5))
C     WRITE(6,*)TLOAD
C
C     EDDY CURRENT RESISTANCE  =  .2 * RS
C     HYSTERESIS RESISTANCE    =  .2 * RS
C
C     GET HYSTERESIS RESISTANCE INDEX
C     INDEX  =  0 OR 1
C     X  =  TIME
C
      CALL HYSINDEX(X,INDEX)
      RH = 0.4*R1*INDEX
C     WRITE(*,*)RH,INDEX
C
C     SET UP DIFFERENTIAL EQUATIONS
C
      DY(1) = VDS-(1.2*RS + RH)*IDS
      DY(2) = VQS-(1.2*RS + RH)*IQS
      DY(3) = Y(5)*N*Y(4)-RR*IDR
      DY(4) = -Y(5)*N*Y(3)-RR*IQR
      DY(5) = 1./J*(TELEC-TLOAD)-CDMPNG/J*Y(5)
```

```
      DY(6) = Y(5)
C
C     OMEGA = DY(6)
C
      DO 10 IU = 1,6
10    DYSAV(IU) = DY(IU)
C
C     PREVENT Y(6) - SHAFT POSITION - FROM GOING PAST 2PI
C
      Y(6) = Y(6)-(IFIX(Y(6)/TWOPI)*TWOPI)
C
C     DY(6) = Y(5) - ANGULAR VELOCITY OF ROTOR IN MECH FRAME - RAD/S
C     DY(5) - ANGULAR ACCELERATION OF ROTOR IN MECH FRAME - RAD/S¢2
C     Y(6) - SHAFT ANGLE - RAD
C
C     COMPUTE ELECTRICAL TORQUE
C
C
      TELEC = 1.5*N*(Y(2)*IDS-Y(1)*IQS)
C
      LDSSTAR = 1./(1./L1 + 1./L2 + 1./LM)
      LQSSTAR = LDSSTAR
      LDM = LDSSTAR*(Y(1)/L1 + Y(3)/L2)
      LQM = LQSSTAR*(Y(2)/L1 + Y(4)/L2)
      IDS = (Y(1)-LDM)/L1
      IQS = (Y(2)-LQM)/L1
      IDR = (Y(3)-LDM)/L2
      IQR = (Y(4)-LQM)/L2
C
C     GENERATE PHASE VOLTAGES
C
      V(1) = VOLTS*SIN(X*OMEGAS)
      V(2) = VOLTS*SIN(X*OMEGAS-2.094395103)
      V(3) = VOLTS*SIN(X*OMEGAS + 2.094395103)
C
      VDS = .577350269*(V(2)-V(3))
      VQS = 1./3.*(2.*V(1)-V(2)-V(3))
C
C     CALCULATE STATOR PHASE CURRENTS
C
      I(1) = IQS
      I(2) = .5*(1.732050808*IDS-IQS)
      I(3) = -.5*(1.732050808*IDS + IQS)
C
C
C     CALCULATE ROTOR PHASE CURRENTS
C
      IR(1) = IQR
      IR(2) = .5*(1.732050808*IDR-IQR)
      IR(3) = -.5*(1.732050808*IDR + IQR)
C
C     ELECTRICAL ANALYSIS COMPLETE
C
      RETURN
      END
C
```

```
      SUBROUTINE DRV0
C
C     THIS ROUTINE INITIALIZES ALL VARIABLES AND INITIAL CONDITIONS
C     FOR THE DERIVATIVE EVALUATION SUBPROGRAM DRV
C
      REAL IQR,IDR,IQS,IDS,L1,L2,LM,LQM,LDM,I,NPOLES,N,J,JS,KS,IR,M,K
      COMMON/ELEC/I(3),V(3),H,TELEC,TLOAD,IR(3),RH
      COMMON/MOTOR/X1,X2,R1,R2,XM,J,FREQ,DFREQ,NPOLES,VRMS,FLT,JS,KS,CS
     +     ,M,C,K,FORCE,CDMPNG
      COMMON/INIT/VOLTS,OMEGAS,OMEGAD,N,VQS,VDS,IQS,IDS,IDR,IQR,
     1  L1,RS,RR,L2,LM
      DATA TWOPI/6.2831853/
C
C     OMEGAD - DESIGN ELECTRICAL FREQUENCY OF INPUT VOLTAGES IN RAD/SE
C           FOR THE PURPOSE OF EVALUATING MOTOR INDUCTANCES
C     VOLTS  - P-P PHASE VOLTAGE
C
      OMEGAS = FREQ*TWOPI
      OMEGAD = DFREQ*TWOPI
      N = NPOLES/2.
C
C     INPUT VOLTAGE FOR PP/RMS AND FOR WYE CONNECTION
C
      VOLTS = VRMS*SQRT(2.)
C
      DO 2 JL = 1,3
      I(JL) = 0.
      IR(JL) = 0.
      V(JL) = 0.
2     CONTINUE
      VQS = 0.
      VDS = 0.
      IQS = 0.
      IDS = 0.
      IDR = 0.
      IQR = 0.
      TELEC = 0.
      L1 = X1/OMEGAD
      RS = R1
      RR = R2
      L2 = X2/OMEGAD
      LM = XM/OMEGAD
      RETURN
      END
C
C
      SUBROUTINE HYSINDEX(TIME,IN)
C
C     FOR HYSTERESIS CALCULATION
C
      INTEGER INDEX,IN
      REAL I,IR
      COMMON/ELEC/I(3),V(3),DELT,TELEC,TLOAD,IR(3),RH
      IN = 0
      NPOLES = 8
```

```
      FREQ = 60.0
C
C     0.46 = POWER FACTOR, ACOS(.46) = 1.093 RAD
C
C     XINDEX = (1/FREQ - 1.093/(120*PI) + TIME)/(1/2*FREQ)
      XINDEX = (1/60-1.093/377. + TIME)*120.
      INDEX = IFIX(XINDEX)
C     WRITE(*,*)INDEX,IN,NPOLES
      IF(INDEX.EQ.2*(INDEX/2)) GO TO 5
      IN = 1
5     RETURN
      END
C
      SUBROUTINE SAVE(NN,TIME,Y,DY)
C
C     COMMON FOR PLOTING DATA
C
      REAL X,I,J,NPOLES,Y(6),DY(6),JS,KS,IR,K,M
      COMMON/MOTOR/X1,X2,R1,R2,XM,J,FREQ,DFREQ,NPOLES,VRMS,FLT,JS,KS,CS
     +     ,M,C,K,FORCE,CDMPNG
      COMMON/ELEC/I(3),V(3),DELT,TELEC,TLOAD,IR(3),RH
      COMMON/PLOTT/X(963,17),DYSAV(6)
      COMMON/TORQ/TLM,TLC,TLK,TLF
      COMMON/PISTON/DISP,VELO,ACCEL
      COMMON/COEFF/COEF,TANBETA
      DATA TWOPI/6.2831853/
C
C
C
C     COLUMN IDENTIFICATION
C
C       1 - TIME - INDEPENDENT VARIABLE
C       2 - I(1) STATOR CURRENTS (AMPS)
C       3 - CORE LOSS (W)
C       4 - COPPER LOSS - STATOR (W)
C       5 - V(1) STATOR VOLTAGES (VOLTS)
C       6 - COPPER LOSS - ROTOR (W)
C       7 - SUM OF LOSSES (W)
C       8 - REAL INSTANTANEOUS POWER
C       9 - ELECTRICAL TORQUE - N-M
C      10 - LOAD TORQUE - N-M
C      11 - SHAFT ACCEL - RAD/S¢2
C      12 - SHAFT VELOCITY - RAD/S
C      13 - SHAFT ANGLE  - RAD
C      14 - SHAFT SPEED - RPM
C      15 - NET OUTPUT TORQUE N-M
C      16 - EDDY CURRENT LOSS (W)
C      17 - HYSTERESIS LOSS (W)
C
      NN = NN + 1
      DO 10 II = 1,17
10    X(1,II) = FLOAT(NN)
C
      THREEI = I(1)*I(1) + I(2)*I(2) + I(3)*I(3)
      X(NN + 1,1) = TIME
      X(NN + 1,2) = I(1)
```

```
      X(NN+1,3)= THREEI*(.2*R1+RH)
      X(NN+1,4)= THREEI*R1
      X(NN+1,5)= V(1)
      X(NN+1,6)= (IR(1)*IR(1)+IR(2)*IR(2)+IR(3)*IR(3))*R2
      X(NN+1,7)= X(NN+1,3)+X(NN+1,4)+X(NN+1,6)
      X(NN+1,8)=I(1)*V(1)+I(2)*V(2)+I(3)*V(3)
      X(NN+1,9)=TELEC
      X(NN+1,10)=TLOAD
      X(NN+1,11)=DYSAV(5)
      X(NN+1,12)=DYSAV(6)
      X(NN+1,13)=Y(6)
      X(NN+1,14)=DYSAV(6)*60./TWOPI
      X(NN+1,15)= (X(NN+1,8)-X(NN+1,7))/(X(NN+1,12)+0.0000001)
      X(NN+1,16)= THREEI*.2*R1
      X(NN+1,17)= THREEI*RH
      WRITE(10,*)X(NN+1,15),TLM,X(NN+1,13)
      WRITE(11,*)X(NN+1,13),X(NN+1,12),X(NN+1,11)
      WRITE(12,*)TLM,TLC,TLK,TLF
      WRITE(13,*)X(NN+1,15)-J*X(NN+1,11)-CDMPING*X(NN+1,12), X(NN+1,13)
      WRITE(14,*)DISP,VELO,ACCEL
      RETURN
      END
C
C
C
C     SUBROUTINE TO CALCULATE LOAD TORQUE
C
      SUBROUTINE LOAD(TLOAD, ANGLE, OMEGA, ALPHA)
      COMMON/MOTOR/X1,X2,R1,R2,XM,J,FREQ,DFREQ,NPOLES,VRMS,FLT,JS,KS,CS
     +     ,M,C,K,FORCE,CDMPNG
      REAL L,MU,M,K,JS,KS,NPOLES,J
      COMMON/PISTON/DISP,VELO,ACCEL
      COMMON/TORQ/TLM,TLC,TLK,TLF
      COMMON/COEFF/COEF,TANBETA
      DATA R,L,MU/0.0254, 0.1016, 0.02/
      CALL LINEAR(ANGLE, OMEGA, ALPHA)
      RSIN=R*SIN(ANGLE)
      RCOS=R*COS(ANGLE)
      TANBETA=RSIN/SQRT(L*L-RSIN*RSIN)
C     TLOAD=(RSIN-RCOS*TANBETA)*(M*ACCEL+C*VELO+K*(DISP-L+R)
C     +     +FORCE+9.81*MU*M)/(1-MU*TANBETA)
      COEF=(RSIN-RCOS*TANBETA)/(1-MU*TANBETA)
      TLM=COEF*(ACCEL+9.81*MU)
      TLC=COEF*VELO
      TLK=COEF*(DISP-L+R)
      TLF=COEF
      TDK=COEF*(DISP-L+R)*K
      TVC=COEF*VELO*C
      TAM=COEF*(ACCEL+9.81*MU)*M
      TPLOD=COEF*FORCE
      TLOAD=TDK+TVC+TAM+TPLOD
      RETURN
      END
C
C     SUBROUTINE TO CALCULATE PISTON DISPLACEMENT, VELOCITY, AND
```

```
C    ACCELERATION
C
     SUBROUTINE LINEAR(ANGLE, OMEGA, ALPHA)
     COMMON/MOTOR/X1,X2,R1,R2,XM,J,FREQ,DFREQ,NPOLES,VRMS,FLT,JS,KS,CS
    +     ,M,C,K,FORCE,CDMPNG
     REAL L,M,K,JS,KS,NPOLES,J
     COMMON/PISTON/DISP,VELO,ACCEL
     DATA R,L/0.0254, 0.1016 /
C    WRITE (*,*)ANGLE,OMEGA,ALPHA
C    WRITE (4,*)ANGLE,OMEGA,ALPHA
     RSIN = R*SIN(ANGLE)
     RCOS = R*COS(ANGLE)
     DISP = -1*RCOS + SQRT(L*L-RSIN*RSIN)
     VELO = DISP*OMEGA*RSIN/(RCOS + DISP)
     ACCEL = (-1*VELO*VELO + DISP*OMEGA*OMEGA*RCOS + DISP*ALPHA*RSIN
    +   + 2*VELO*OMEGA*RSIN)/(RCOS + DISP)
     RETURN
     END
C
C    SET YTERM 1.4 IN TEKTRONIX MODE
C
     SUBROUTINE DISPLAY
     REAL IR,IJ
     COMMON/PLOTT/X(963,17),DYSAV(6)
     COMMON/ELEC/IJ(3),V(3),DELT,TELEC,TLOAD,IR(3),RH
     DIMENSION T1(964),T2(964),T3(945),T4(945)
     WRITE(6,*) CHAR(39),CHAR(173),CHAR(76),CHAR(241),CHAR(136)
     DO 15 I =  1,962
15   T1(I) = X(I + 1,1)
11   WRITE(*,12)
12   FORMAT(' INPUT A NUMBER 1-16, 17 EXIT , THEN PRESS C AND RETURN')
     READ(*,*)IRRPLY
     GO TO (100,200,300,400,500,600,700,800,900,
    +   1000,1100,1200,1300,1400,1500,1600,1700),IRRPLY
100  DO 110 I =  1,962
110  T2(I) = X(I + 1,2)
111  CALL PLOTS( 0,0, 50 )
     CALL SCALE(T1,8.,962,1)
     CALL SCALE(T2,6.,962,1)
     CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
     CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
     CALL AXIS(0.5,0.5,'1.I AMP',7,6.,90.,T2(963),T2(964))
     CALL AXIS(8.5,0.5,'1.I AMP',-7,6.,90.,T2(963),T2(964))
     CALL PLOT(0.5,0.5,-3)
     CALL LINE(T1,T2,962,1,0,0)
     CALL PLOT( 0.0, 0.0, 999 )
     GO TO 11
200  DO 210 I =  1,962
210  T2(I) = X(I + 1,3)
211  CALL PLOTS( 0,0, 50 )
     CALL SCALE(T1,8.,962,1)
     CALL SCALE(T2,6.,962,1)
     CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
     CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
     CALL AXIS(0.5,0.5,'2.CORE LOSS W',13,6.,90.,T2(963),T2(964))
     CALL AXIS(8.5,0.5,'2.CORE LOSS W',-13,6.,90.,T2(963),T2(964))
```

```
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
  300  DO 310 I = 1,962
  310  T2(I)=X(I+1,4)
  311  CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'3.STATOR LOSS W',15,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'3.STATOR LOSS W',-15,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
  400  DO 410 I = 1,962
  410  T2(I)=X(I+1,5)
  411  CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'4.INPUT VOLT',12,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'4.INPUT VOLT',-12,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
  500  DO 510 I = 1,962
  510  T2(I)=X(I+1,6)
  511  CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'5.ROTOR I¢2 LOSS W',18,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'5.ROTOR I¢2 LOSS W',-18,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
  600  DO 610 I = 1,962
  610  T2(I)=X(I+1,7)
  611  CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'6.TOTAL LOSS W',14,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'6.TOTAL LOSS W',-14,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
```

```
      GO TO 11
  700  DO 710 I = 1,962
  710  T2(I)=X(I+1,8)
  711  CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'7.INPUT POWER W',15,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'7.INPUT POWER W',-15,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
  800  DO 810 I = 1,962
  810  T2(I)=X(I+1,9)
  811  CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'8.TELEC N-M',11,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'8.TELEC N-M',-11,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
  900  DO 910 I = 1,962
  910  T2(I)=X(I+1,10)
  911  CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'9.TLOAD N-M',11,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'9.TLOAD N-M',-11,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
 1000  DO 1010 I = 1,962
 1010  T2(I)=X(I+1,11)
 1011  CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'10.SHAFT ALPHA R/SS',19,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'10.SHAFT ALPHA',-14,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
 1100  DO 1110 I = 1,962
 1110  T2(I)=X(I+1,12)
 1111  CALL PLOTS( 0,0, 50 )
```

```
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'11.SHAFT OMEGA R/S',18,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'11.SHAFT OMEGA R/S',-18,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
 1200 DO 1210 I = 1,962
 1210 T2(I)=X(I+1,13)
 1211 CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'12.SHAFT ANGLE',14,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'12.SHAFT ANGLE',-14,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
 1300 DO 1310 I = 1,962
 1310 T2(I)=X(I+1,14)
 1311 CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'13.ROTOR RPM',12,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'13.ROTOR RPM',-12,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
 1400 DO 1410 I = 1,943
      T3(I)=X(I+20,1)
 1410 T4(I)=X(I+20,15)
 1411 CALL PLOTS( 0,0, 50 )
      CALL SCALE(T3,8.,943,1)
      CALL SCALE(T4,6.,943,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T3(944),T3(945))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T3(944),T3(945))
      CALL AXIS(0.5,0.5,'14.NET TOUT N-M',15,6.,90.,T4(944),T4(945))
      CALL AXIS(8.5,0.5,'14.NET TOUT N-M',-15,6.,90.,T4(944),T4(945))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T3,T4,943,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
 1500 DO 1510 I = 1,962
 1510 T2(I)=X(I+1,16)
 1511 CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
```

```
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'15.EDDY C LOSS W',16,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'15.EDDY C LOSS W',-16,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
1600  DO 1610 I = 1,962
1610  T2(I)=X(I+1,17)
1611  CALL PLOTS( 0,0, 50 )
      CALL SCALE(T1,8.,962,1)
      CALL SCALE(T2,6.,962,1)
      CALL AXIS(0.5,0.5,'TIME SEC',-8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,6.5,'TIME SEC',8,8.,0.,T1(963),T1(964))
      CALL AXIS(0.5,0.5,'16.HYSTERESIS L W',15,6.,90.,T2(963),T2(964))
      CALL AXIS(8.5,0.5,'16.HYSTERESIS L W',-15,6.,90.,T2(963),T2(964))
      CALL PLOT(0.5,0.5,-3)
      CALL LINE(T1,T2,962,1,0,0)
      CALL PLOT( 0.0, 0.0, 999 )
      GO TO 11
C
C     REMOVE YTERM 1.4 FORM TEKTRONIX MODE
C
1700  WRITE(6,*) CHAR(39),CHAR(24)
      RETURN
      END
```

A.3 Execution file of Solution Program for IBM/VM

```
/* PROFILE FOR SOLUTION PROGRAM EASIER FORTRAN */
/* ENABLES THE USE OF IMSL SUBROUTINE */
  TRACE ON
 'GLOBAL TXTLIB VSF2FORT ESVVLIB IMSLSP IMSL1 IMSL2 CMSLIB'
 'GLOBAL LOADLIB VSF2LOAD'
/* DEFINE INPUT AND OUPPUT DISKS*/
 ' FILEDEF 10 DISK TOUTJA DAT G'
 ' FILEDEF 11 DISK TLMCKF DAT G'
 ' FILEDEF 40 DISK RESULT DAT G'
 'LOAD EASIER (NOMAP'
 'START'
```

A.4 Solution Program

```
C************* MAIN PROGRAM ****************************************
C     TIME DOMAIN SIGNAL
C     TNET : NET OUTPUT TORQUE
C     TLMP : LOAD TORQUE MASS CONTRIBUTION
C     TLC  : LOAD TORQUE DAMPING CONTRIBUTION
```

```
C     TLK  : LOAD TORQUE STIFFNESS CONTRIBUTION
C     TLF  : LOAD TORQUE PRELOAD FORCE CONTRIBUTION
      REAL TNET(160),ATNET(11),BTNET(11),CTNET(11),TIME(160)
      REAL TLMP(160),ATLMP(11),BTLMP(11),CTLMP(11)
      REAL TLC(160),ATLC(11),BTLC(11),CTLC(11)
      REAL TLK(160),ATLK(11),BTLK(11),CTLK(11)
      REAL TLF(160),ATLF(11),BTLF(11),CTLF(11)
      REAL TLC1(160),TLK1(160),TLF1(160)
      COMMON/QNET/ATNET,BTNET,CTNET
      COMMON/QLMP/ATLMP,BTLMP,CTLMP
      COMMON/QTLC/ATLC,BTLC,CTLC
      COMMON/QTLK/ATLK,BTLK,CTLK
      COMMON/QTLF/ATLF,BTLF,CTLF
C     INPUT TOTAL NUMBER OF TIME DOMAIN DATA
      WRITE(*,*) ' INPUT DATA NUMBER, MAX = 160 '
      READ(*,*) NPTS
      PERIOD = 0.0673
      DX = PERIOD/(FLOAT(NPTS-1))
      PI = 4.0*ATAN(1.0)
      NH = 11
      DO 1 I =  1,NPTS
      TIME(I) = (I-1)*PERIOD/FLOAT(NPTS-1)
      READ(10,*)TNET(I)
      READ(11,*)TLC(I),TLF(I)
1     CONTINUE
      CALL DFT (NPTS,NH,DX,TIME,TNET,ATNET,BTNET,CTNET)
      CALL DFT (NPTS,NH,DX,TIME,TLMP,ATLMP,BTLMP,CTLMP)
      CALL DFT (NPTS,NH,DX,TIME,TLC,ATLC,BTLC,CTLC)
      CALL DFT (NPTS,NH,DX,TIME,TLK,ATLK,BTLK,CTLK)
      CALL DFT (NPTS,NH,DX,TIME,TLF,ATLF,BTLF,CTLF)
      CALL LSTSQR
      STOP
      END
C************* D.F.T. SUBROUTINE cf. Brigham[12]**********************
      SUBROUTINE DFT(NPTS,NH,DX,X,Y,A,B,C)
C
      INTEGER NPTS,NH
      REAL X(601),Y(601),A(51),B(51),C(100),DX
      REAL RNH
      REAL PI
C
      PI = 4.0*ATAN(1.0)
      TIME = DX*FLOAT(NPTS-1)
C
C     PERFORM DFT ON DATA
C     CALCULATE An'S
      DO 20 J = 1,NH
      N = J-1
      A(J) = (Y(1)*COS(2.*N*PI*X(1)/TIME) +
     $ Y(NPTS)*COS(2.*N*PI*X(NPTS)/TIME))/2.
      DO 30 I = 2,NPTS-1
      A(J) = A(J) + Y(I)*COS(2.*N*PI*X(I)/TIME)
   30 CONTINUE
      A(J) = A(J)*DX*2./TIME
```

```
  20 CONTINUE
C
C CALCULATE Bn'S
      DO 40 J = 2,NH
      N = J-1
      B(J) = (Y(1)*SIN(2.*N*PI*X(1)/TIME) +
     $ Y(NPTS)*SIN(2.*N*PI*X(NPTS)/TIME))/2.
      DO 50 I = 2,NPTS-1
      B(J) = B(J) + Y(I)*SIN(2.*N*PI*X(I)/TIME)
   50 CONTINUE
      B(J) = B(J)*DX*2./TIME
   40 CONTINUE
C
C CALCULATE Cn'S
      RNH = NH
      C(1) = A(1)
      DO 76 I = 2,NH
      C(I) = SQRT(A(I)*A(I) + B(I)*B(I))
   76 CONTINUE

      RETURN
      END
C*************** END OF D.F.T. SUBROUTINE ***************************
C*************** LEAST SQUARES SUBROUTINE ***************************
      SUBROUTINE LSTSQR
      PARAMETER (NRA = 5,NCA = 4,LDA = NRA)
      REAL A(LDA,NCA),B(NRA),X(NCA),RES(NRA),TOL
      REAL ATLC(11),BTLC(11),CTLC(11),ATLK(11),BTLK(11),CTLK(11),
     $ ATLM(11),BTLM(11), CTLM(11),ATNET(11),BTNET(11),CTNET(11),
     $ ATLF(11),BTLF(11), CTLF(11)
      CTLKMON/QNET/ATNET,BTNET,CTNET
      CTLKMON/QLMP/ATLM,BTLM,CTLM
      CTLKMON/QTLC/ATLC,BTLC,CTLC
      CTLKMON/QTLK/ATLK,BTLK,CTLK
      CTLKMON/QTLF/ATLF,BTLF,CTLF
      TOL = 0.00001
C
C     INPUT MATRIX VALUE:
C
C   | A(1,1)  A(1,2)  A(1,3)  A(1,4)      M     B(1)
C   | A(2,1)  A(2,2)  A(2,3)  A(2,4)      C     B(2)
C   | A(3,1)  A(3,2)  A(3,3)  A(3,4)      K     B(3)
C   | A(4,1)  A(4,2)  A(4,3)  A(4,4)      F     B(4)
C   | A(5,1)  A(5,2)  A(5,3)  A(5,4)            B(5)
C
      A(1,1) = ATLM(1)
      A(1,2) = ATLC(1)
      A(1,3) = ATLK(1)
      A(1,4) = ATLF(1)
      B(1) = ATNET(1)
      A(2,1) = BTLM(2)
      A(2,2) = BTLC(2)
      A(2,3) = BTLK(2)
      A(2,4) = BTLF(2)
      B(2) = BTNET(2)
      A(3,1) = BTLM(3)
```

```
      A(3,2) = BTLC(3)
      A(3,3) = BTLK(3)
      A(3,4) = BTLF(3)
      B(3) = BTNET(3)
      A(4,1) = BTLM(4)
      A(4,2) = BTLC(4)
      A(4,3) = BTLK(4)
      A(4,4) = BTLF(4)
      B(4) = BTNET(4)
      A(5,1) = BTLM(5)
      A(5,2) = BTLC(5)
      A(5,3) = BTLK(5)
      A(5,4) = BTLF(5)
      B(5) = BTNET(5)
      CALL LSQRR (NRA,NCA,A,LDA,B,TOL,X,RES,KBASIS)
      CALL UMACH (2,NOUT)
      WRITE (NOUT,*)' KBASIS = ',KBASIS
      CALL WRRRN ('X',1,NCA,X,1,0)
      CALL WRRRN ('RES',1,NRA,RES,1,0)
      WRITE (40,*) 'MASS M    DAMPING C      STIFFNESS K    PRELOAD F'
      WRITE (40,*) X(1),X(2),X(3),X(4)
      RETURN
      END
C********** END OF LEAST SQUARES SUBROUTINE **********************
```

Vita

Yun-chung Tang was born on September 29, 1962, in Tainan, Taiwan, R.O.C. He completed his high school education in Taipei, Taiwan. In June, 1985, he graduated from National Taiwan University with a bachelor's degree in Mechanical Engineering. The following two years, he served in the Taiwan Army as a platoon leader in charge of four tanks. After discharge from the Army, he worked for two years for two companies as a salesman and as a service engineer. In September 1989 he enrolled in the Master's program in Mechanical Engineering at Virginia Tech.

Mr. Tang will complete the requirements for the degree of Master of Science in Mechanical Engineering in October, 1991.