# ANALYSIS OF DYNAMIC CHARACTERISTICS OF FLEXIBLE ROBOT ARM USING SYMBOLIC MANIPULATION

**Mustafa T. Hussein**
**Department of mechanical Engineering**
**University of Babylon**

**Essam Z. Fadhel**
**Department of mechanical Engineering**
**University of Babylon**

**Abstract**

This paper presents the symbolic analysis of the dynamic model of flexible robot arm. A planar single flexible link is considered. The finite element method is used for dynamic modeling of the system; the link is modeled as single finite element to simplify the problem. A symbolic algorithm is developed using MatLab program, general symbolic solution describing the dynamic behavior of single link flexible manipulator is obtained; including two types of boundary conditions, payload and frictional terms. The effect of physical parameters of the manipulator on the characteristics of the system and the impact of their variation on the transfer function features of the system are clearly shown by the symbolic algorithm presented in this paper.

**Keywords**: Flexible Robot, Symbolic Modeling, Transfer Function.

# تحليل الخصائص الديناميكيةِ للذراع الآلي المرنِ باستخدام طريقة المعاملة الرمزية

**مصطفى تركي حسين**
**قسم الهندسة الميكانيكية**
**جامعة بابل**

**عصام زهير فاضل**
**قسم الهندسة الميكانيكية**
**جامعة بابل**

**الخلاصة**

في هذا البحث تم تقديم التحليل الرياضي لمعادلة الحركة لذراع الإنسان الآلي المرنة باستخدام طريقة الوصف الرمزي لمعادلة الحركة. تم اخذ نموذج يتكون من ذراع مفردة، ويكون التشوه فيها بمستوي واحد للدراسة. استخدمت طريقة العناصر المحددة (FEM) للحصول على معادلة الحركة لذراع الإنسان الآلي المرنة، تم معاملة الذراع على إنها قطعة واحدة (SFE) وذلك لتسهيل التمثيل الرياضي للمسألة. تم تطوير أسلوب تمثيل رياضي رمزي باستخدام برنامج (ماتلاب) لإنتاج حل رياضي عام رمزي لذراع الإنسان الآلي المرنة بمستوي واحد لوصف السلوك الديناميكي للذراع، إضافة إلى إن هذا الحل يتضمن إضافة الأحمال أو مؤثرات الاحتكاك ولنوعين من الشروط الحدية. إن تأثير العوامل الفيزيائية للذراع المرنة على خصائص كل من معادلات الحركة ومعادلات السيطرة (T.F.) قد تم توضيحه من خلال اشتقاق المعادلات الرمزية.

**Nomenclatures**

$\phi$ : the shape function.

$\theta_1$ : the slope at the base end of the link.

$v_2$ : the vertical deflection at the end of the link.

$\theta_2$ :the slope at the free end of the link.

$\tau$ :the generalized forces vector.

$\mathbf{K}_T$ : the generalized stiffness matrix.
$\mathbf{M}_T$ :The generalized mass matrix.
**N** : gear ratio.
$x$ : the distance along the beam.
t: time.
FEM: Finite Element Method.
RHP: Right half Plane (S-plane).

## Introduction

Lightweight mechanical structures are expected to improve performance of robot manipulators with typically low payload-to-arm weight ratio. The ultimate goal of such robotic designs is to achieve fast and dexterous motion as opposed to slow and bulky motion of conventional industrial robots. Although the transfer to applications of advanced research findings in this field is still in its infancy, we believe that the realization of effective lightweight robots may prove very promising in a number of innovative areas including manipulation with very long arms, teleoperation, and space robotics (Alessandro De Luca 1991).

In order to fully exploit the potential advantages offered by lightweight robot arms, one must explicitly consider the effects of structural link flexibility and properly deal with (active and/or passive) control of vibrational behavior. In this context, it is highly desirable to have an explicit, complete, accurate dynamic model at disposal. The model should be explicit to provide a clear understanding of dynamic interaction and coupling effects, to be useful for control design, and to guide reduction and/or simplification based on terms relevance.

First the symbolic forms of the equations similar to rigid manipulator equations would be very useful in control studies since researchers are already familiar in dealing with rigid manipulator control problems second even when the general form of the equations is available in symbolic form , customized expansion of these equations is needed for dynamic-model-based real-time control implementations so that the cancellations are already taken care of and the computational requirements in real time is minimized. Expansion of the symbolic equations by hand for a given manipulator my be very tedious and error prone. Therefore, it is desirable to also automate the symbolic expansion using one of the symbolic manipulation programs (SMP, MACSYMA, and MATHMATICA). These capabilities are available for rigid manipulators as a result of studies reported in (Sabri 1992).

Several techniques have previously been developed for modeling of flexible robot manipulators. These include solving the partial differential equation (PDE) characterizing the dynamic behavior of a flexible robot manipulator using approximate modes (Park and Asada 1990) and numerical analyses**.** The modeling results show that flexible manipulators have zeros in the right half **s** plane (RHP), thus characterizing a non-minimum phase system (Tokhi et. al. 1999). This further makes the control of a flexible manipulator complicated and imposes limitations on the performance of the system. Therefore, this aspect needs to be further studied, as it will help with the process of controller design.

The aim of the work presented in this paper is to investigate and analyze non-minimum phase characteristic of single link flexible robot using a symbolic manipulation in matlab program. In this work, the finite element (FE) method is used for dynamic modeling of the system. Using the proposed symbolic manipulation approach, a general solution describing the dynamic behavior of the manipulator can be obtained using the matlab program; the transfer function of the system which SIMO (single input multi output) system was obtained by using the same symbolic manipulation.

## Problem formulation

The flexible body dynamics of the system refer to the deflection of the beam from the expected rigid body position. The equations for this motion can be determined using the Lagrange equations and the assumed modes method (Mustafa et. al. 2009). If the deflection is assumed to be the sum of the generalized coordinates multiplied by shape functions, the equations can be determined by computing the generalized mass, generalized stiffness, generalized force, and generalized moment. The generalized coordinates and shape functions can be determined using the finite-element method FEM. This is less labor intensive than solving the Euler-Bernoulli beam equation, which is a fourth order partial differential equation (Hastings and Book 1986).

The dynamic equation of motion of the system are derived using computer program new functions which is derived symbolic or numeric form of mass, stiffness, and load matrices. two different beam models was used in the matlab functions for modeling the flexible arm portion of the system a clamped-free model, and a pinned-free model.

**Figure 1-a** shows single link flexible with pinned free boundary conditions, b shows single link flexible with clammped free boundary conditions. For simplicity, the beam is modeled as a single finite element. This is adequate since the system behavior is dominated by the first mode, which is a relative simple shape function (Thomson 1998).

If a pinned-beam is modeled as a single finite element, there are three generalized (independent) coordinates: slope of the beam at the hub, $\theta_1$ the tip deflection, $v_2$ and the tip slope $\theta_2$. The deflection of the beam at any point along its length is then:

$$y(x,t) = \theta_1(t)\phi_1(x) + v_2(t)\phi_2(x) + \theta_2(t)\phi_3(x) \quad (1)$$

Where $\phi_1$ , $\phi_2$ and $\phi_3$ are the shape functions and $x$ is the distance along the beam. $\theta_1, v_2$ and $\theta_2$ are coordinates of time only, and $\phi_1$ , $\phi_2$ and $\phi_3$ are functions of space only (distance, $x$).

For the clamped-free beam shown in **Figure 1-b** (modeled as a single finite element, there are only two generalized (independent) coordinates: the tip deflection, $v_2$ and the tip slope $\theta_2$. The deflection of the beam at any point along its length is that:

$$y(x,t) = v_2(t)\phi_1(x) + \theta_2(t)\phi_2(x) \quad (2)$$

Where $\phi_1$ and $\phi_2$ are the shape functions and $x$ is the distance along the beam. Note that $v_2$ and $\theta_2$ are functions of time only, $\phi_1$ and $\phi_2$ are functions of space only.

The generalized mass, $\mathbf{M}_T$ and generalized stiffness, $\mathbf{K}_T$ are determined from these shape functions. Knowing these values as well as the generalized forces $\tau$ the equation for the flexible body system is:

$$\mathbf{M}_T \begin{bmatrix} \ddot{\theta}_m \\ \ddot{v}_2 \\ \ddot{\theta}_2 \end{bmatrix} + \mathbf{B}_T \begin{bmatrix} \dot{\theta}_m \\ \dot{v}_2 \\ \dot{\theta}_2 \end{bmatrix} + \mathbf{K}_T \begin{bmatrix} \theta_m \\ v_2 \\ \theta_2 \end{bmatrix} = \mathbf{Q}\tau \quad (3)$$

Where $\theta_1 = \theta_m / N$ , the form of this equation is depending on the type of boundary conditions pinned or clammped (Mustafa et. al. 2009).

## Symbolic algorithm

In this section, the procedure for the development of the symbolic algorithm is presented. To study the relation between system parameters and non-minimum phase features, a system transfer function based on a single element FE simulation of the system should be obtained and studied.

Moreover, the effect of hub, payload, and friction forces are taken into accounts to show the effect of nonlinear parameters on the dynamic equations and transfer function of the system.

As stated earlier using the FE method to solve dynamic problems of the flexible manipulator gives elemental mass matrices for the *n* elements, which are assembled to obtain system mass and stiffness matrices, **M** and **K**, and used in the Lagrange equation to obtain the dynamic equation of the flexible manipulator as:

$$\mathbf{M}_T\begin{bmatrix}\ddot{\theta}_m\\ \ddot{v}_2\\ \ddot{\theta}_2\end{bmatrix}+\mathbf{B}_T\begin{bmatrix}\dot{\theta}_m\\ \dot{v}_2\\ \dot{\theta}_2\end{bmatrix}+\mathbf{K}_T\begin{bmatrix}\theta_m\\ v_2\\ \theta_2\end{bmatrix}=\mathbf{Q}\tau \tag{4}$$

The matrix differential equation in equation (4) can be represented in a state-space form as(Mustafa et. al. 2009).:

$$\begin{aligned}\dot{\mathbf{x}}&=\mathbf{Ax}+\mathbf{Bu}\\ \mathbf{y}&=\mathbf{Cx}\end{aligned} \tag{5}$$

Where

$$\mathbf{x}=\begin{bmatrix}\theta_m\\ v_2\\ \theta_2\\ \dot{\theta}_m\\ \dot{v}_2\\ \dot{\theta}_2\end{bmatrix},\quad \mathbf{y}=\begin{bmatrix}\theta_1\\ v_2\\ \theta_2\\ \dot{\theta}_m\\ \dot{v}_2\\ \dot{\theta}_2\end{bmatrix},\ \textit{and}\ u=\tau=N\tau_a$$

$$\mathbf{A}=\begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{I}\\ \mathbf{M}_T^{-1}\mathbf{K}_T & \mathbf{M}_T^{-1}\mathbf{B}_T\end{bmatrix},\mathbf{B}=\begin{bmatrix}0\\ \mathbf{M}_T^{-1}\mathbf{Q}\end{bmatrix},\ \text{and}\ \mathbf{C}=\begin{bmatrix}1/N & \mathbf{0}_{1\times5}\\ \mathbf{0}_{5\times1} & \mathbf{I}\end{bmatrix} \tag{6}$$

Where $\mathbf{x},\mathbf{y},\mathbf{u},\mathbf{A},\mathbf{B},\mathbf{and}\,\mathbf{C}$ are the state vector, output variable, input variable, state matrix, input matrix, and the output matrix; the equations above are for the pinned boundary conditions, the vectors and matrices are derived in detailed in (Mustafa et. al. 2009). Solving the state-space representation gives the system transfer function, these state space equations are also depends on the types of boundary conditions whether it clammped or pinned, and may or may not the existence of payload or frictional terms.

So in the MatLab program new functions are build to generate the dynamic equations of motion and the state space form of single link flexible robot arm; these functions works depends on the FEM procedure to find the mathematical model, by calculating the mass, stiffness, and nonlinear terms matrices. These functions are works in group which all function should be in the same path in the program to use it; the functions are summarized in the table No. 1.

## Analysis Of The System Ccharacteristics

In this section, the system transfer function obtained from the state space form of the dynamic model which is derived in the previous section. This transfer function is assessed to investigate the non-minimum phase nature of the flexible manipulator system. This involves investigating numerator of the transfer function and locations of system zeros. A system is a non-minimum phase if one or more zeros lie on the (RHP).

The state space symbolic form of the dynamic model presented depends on the four above program functions and from this symbolic form can easily find the transfer function of the system :

$$G(s)=\frac{Y(s)}{U(s)}=C(sI-A)^{-1}B \tag{7}$$

where **s** is the Laplace variable and **I** is the identity matrix. Substituting for the matrices **A** , **B,** and **C** from equation (6) into equation (7) and simplifying yields system transfer function with the FE simulation incorporating a single element. The MatLab functions(mass, stiffness, mode_shape) is used by the function (dyn_m) to create the dynamic model of single link flexible robot arm, the introduction of this function is shown in **Figure 2**.

**Figure 2** present the command window of the matlab program, where the help of the function is showed, and the **Figure 3** below shown the symbolic form dynamic model of the clamped-payload B.C.'s of flexible robot arm.

From these matrices the state space form of the dynamic model can be obtained using the code below :

```
>>syms L m EI mp
[mt,kt,bt]=dyn_m1(0);
im=inv(mt);Q=[1 0 0];
A_21=im*kt;A_22=im*bt;
A=[zeros(3),eye(3);A_21,A_22];b_2=im*Q';
B=[0 0 0 b_2']';
C(1,1)=1/N;C(1,2:6)=0;C(2:6,1)=0;C(2:6,2:6)=eye(5);
```

This code using dyn-m1 function which is neglect the gear ratio, inertia of motor and shaft gear, and the viscous friction; to derived the state apace form of the dynamic model (A, B, C) matrices as shown in the **Figure 4**.

And now by use the equation (7) the transfer function of the system can be derived using the matrices above. The transfer function of this system which is SIMO system is fairly complicated due to the number of output from the system of single input, the transfer function in this example is chosen between input torque $(\tau)$ , and the output $(\theta)$.

$$G(s)=\frac{\theta(s)}{\tau(s)}=\frac{300(mL+12m_p)}{(s^2m-EI)*L^3*(mL+15m_p)} -\frac{900EI*m_p}{(s^2mL+15s^2m_p-EI*L)*(s^2m-EI)*L^2*(mL+15m_p)} \tag{8}$$

The system transfer function obtained above is assessed to investigate the non_minimum phase nature of the flexible manipulator system. This involves investigating numerator and denominator of the transfer function and locations of system poles and zeros. A system is a non-minimum phase if one or more zeros lie on the RHP; the pole-zero map is shown in **Figure 5** below for the different value of system parameters, the location of the poles and zeros of the system was changed with decreasing the values of the mass per unit length, length of the link, and payload with the same cross section area.

## Conclusion

A symbolic manipulation for investigation of non-minimum phase characteristic of a flexible robot manipulator has been developed using the matlab program. It has been demonstrated that the algorithm gives a symbolic or numeric general solution of the system dynamic model. The way using of program is presented through the symbolic example. The transfer function of the system was derived symbolically to explain the effect the system parameters. Furthermore, relation between physical parameters, location of system zeros and non-minimum phase characteristics of the system has been obtained and discussed.

**References**

Alessandro De Luca and Bruno Siciliano, "Closed-Form Dynamic Model of Planar Multilink Lightweight Robots", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, Vol. 21, No. 4, 1991.

Hastings and Book, "Verification of a Linear Dynamic Model for Flexible Robotic Manipulators" Proceedings IEEE International Conference on Robotics and Automation, pp. 1024- 1029, 1986.

J. H. Park and H. Asada, "Design and control of minimum-phase flexible arms with torque transmission mechanisms", Proceedings of International Conference of Robotics and Automation, Cincinnati, pp. 1790-1795, 1990.

John h. Mathews , Kurtis D. Fink," Numerical Methods Using Matlab", third edition , prentice hall 1999.

M. O. Tokhi, Z. Mohamed, and A. W. I. Hashim," Application of Symbolic Manipulation to the Analysis of Dynamic Characteristics of a Flexible Robot Manipulator", Institution of Electrical Engineer, 1999.

Mustafa et. al., "Mathematical Modeling of Flexible Robot Arm Using Finite Element Method", Journal of Babylon University\ Engineering, Vol. 17 , No.2.

Sabri Cetinkunt and Babu Ittoop, "computer-Automated Symbolic modeling of Dynamics of Robotics Manipulators with Flexible Links", IEEE Transaction on Robotics and Automation, Vol. 8, No.1, 1992.

Thomson, T. and Dahleh, M.D. "Theory of Vibration with Application", Prentice-Hall, Inc, 1998.

**Table 1. Matlab Functions**

| No. | Function | Description |
|---|---|---|
| 1. | dyn_m | this function create dynamic model for single link flexible robot arm. |
| 2. | dyn_m1 | this function create dynamic model for single link flexible robot arm without including and inertial terms of shaft and arm. |
| 3. | mass | function mass generate mass matrix of flexible robot arm. |
| 4. | stiffness | function stiffness generate stiffness matrix of flexible robot arm |
| 5. | mode_shape | This function generates the mode shape of the flexible link. |

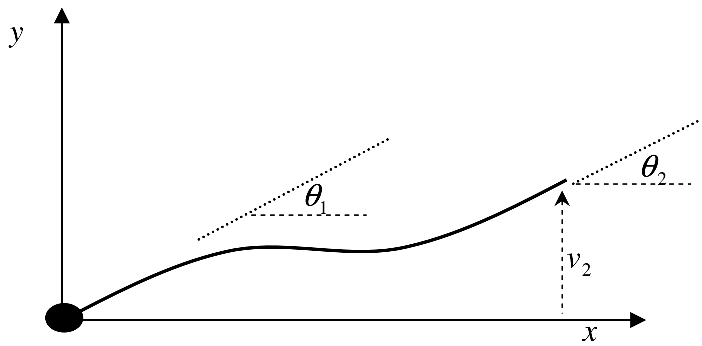


Figure 1-a pinned free beam deflections in the x-y plane.

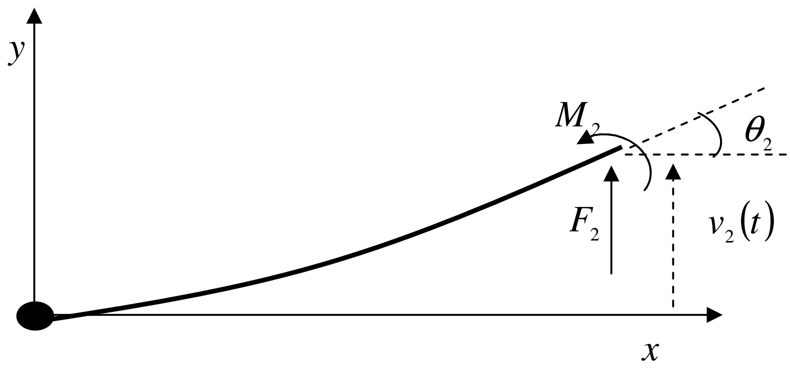


Figure 1-b clamped free beam deflection in the x-y plane.

```
>> help Dyn_m
 This function create dynamic model for single link flexible robot arm.
 The mass, stiffness,and damping matrices of model was created.
 The modeling is based on the finite element method (FEM).
 The value of c must be '0' for pinned or '1' for clamped B.C.'s.
 The values of c,L,m,EI are type of boundary conditions,length of the link,
 mass of the link,and Modulus of elasticity multiplied by moment of inertia.
 The values of N,Jm,Jg2,bm,bs are the gear ratio,the moment of inertia of
 the motor, the moment of inertia of arm gear, the viscous friction of the motor
 the viscous friction of the gear arm.
  Ex.
 [m,k,b]=dyn_m(0);
 this example present the symbolic form dynamic model of pinned B.C.'s
```

Figure 2. Matlab Command window dyn_m function help

```
>> [m,k,b]=dyn_m(0)

m =

[ 1/105*m*L^3+Jm*N+Jg2/N,       13/420*m*L^2,       -1/140*m*L^3]
[        13/420*m*L^2,        13/35*m*L+mp,       -11/210*m*L^2]
[        -1/140*m*L^3,       -11/210*m*L^2,        1/105*m*L^3]



k =

[   1/105*EI*L^3,  13/420*EI*L^2,  -1/140*EI*L^3]
[  13/420*EI*L^2,     13/35*EI*L, -11/210*EI*L^2]
[  -1/140*EI*L^3, -11/210*EI*L^2,   1/105*EI*L^3]



b =

[ N*bm+bs/N,        0,        0]
[         0,        0,        0]
[         0,        0,        0]
```

Figure 3. Matlab Command window dynamic equation matrices

```
>> pretty(B)

                    [          0          ]
                    [                    ]
                    [          0          ]
                    [                    ]
                    [          0          ]
                    [                    ]
                    [      m L + 12 mp    ]
                    [300 -----------------]
                    [       3             ]
                    [   m L  (m L + 15 mp)]
                    [                    ]
                    [         30          ]
                    [   ---------------    ]
                    [   L (m L + 15 mp)    ]
                    [                    ]
                    [    13 m L + 90 mp   ]
                    [30 ----------------- ]
                    [      3              ]
                    [   m L  (m L + 15 mp) ]
>> C

C =

     1     0     0     0     0     0
     0     1     0     0     0     0
     0     0     1     0     0     0
     0     0     0     1     0     0
     0     0     0     0     1     0
     0     0     0     0     0     1
```

```
>> pretty(A)

        [0 , 0 , 0 , 1 , 0 , 0]

        [0 , 0 , 0 , 0 , 1 , 0]

        [0 , 0 , 0 , 0 , 0 , 1]

        [     (m L + 12 mp) EI   13    L EI           (13 m L + 90 mp) EI
        [20/7 ---------------- + -- ----------- - 3/14 ------------------- ,
        [     m (m L + 15 mp)    14 m L + 15 mp         m (m L + 15 mp)

             (m L + 12 mp) EI           EI           (13 m L + 90 mp) EI
        65/7 ----------------- + 78/7 ----------- - 11/7 ------------------- ,
             m L (m L + 15 mp)        m L + 15 mp        m L (m L + 15 mp)

               (m L + 12 mp) EI          L EI          (13 m L + 90 mp) EI
        - 15/7 ---------------- - 11/7 ----------- + 2/7 ------------------- ,
               m (m L + 15 mp)         m L + 15 mp       m (m L + 15 mp)

               ]
        0 , 0 , 0]
               ]

        [      L EI                  ]
        [0 , ----------- , 0 , 0 , 0 , 0]
        [    m L + 15 mp                ]

        [    (13 m L + 90 mp) EI           L EI       15 (13 m L + 48 mp) EI
        [2/7 ------------------- + 13/4 ----------- - -- ------------------- ,
        [      m (m L + 15 mp)          m L + 15 mp   28   m (m L + 15 mp)

        13 (13 m L + 90 mp) EI          EI        55 (13 m L + 48 mp) EI
        -- ------------------- + 39 ----------- - -- ------------------- ,
        14  m L (m L + 15 mp)       m L + 15 mp   14  m L (m L + 15 mp)

               (13 m L + 90 mp) EI          L EI
        - 3/14 ------------------- - 11/2 -----------
                 m (m L + 15 mp)          m L + 15 mp

               (13 m L + 48 mp) EI           ]
         + 5/7 ------------------- , 0 , 0 , 0]
                 m (m L + 15 mp)             ]
>>
```

Figure 4. Matlab command window State-Space matrices.

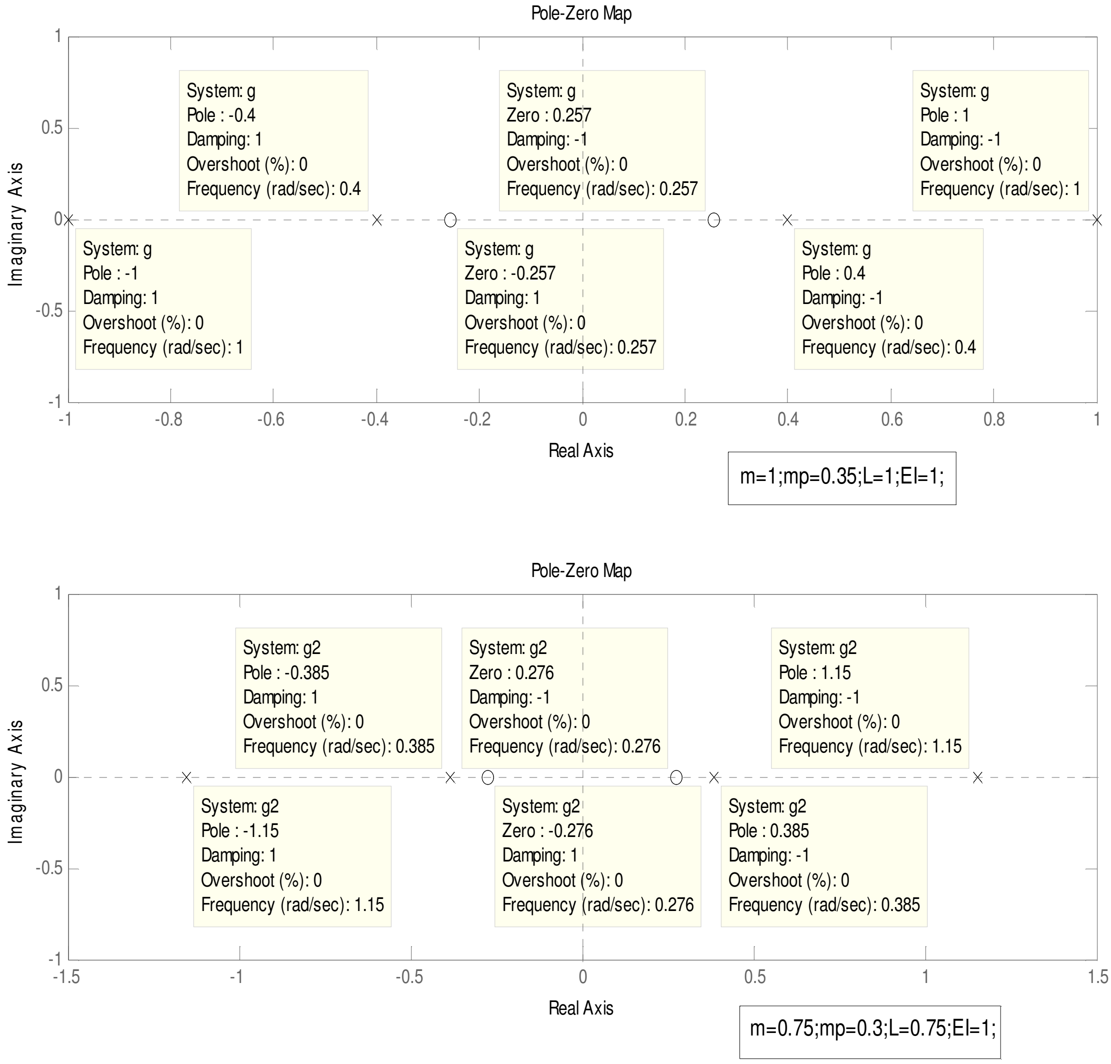


Figure 5. pole-zero-map for the system