Precise Performance Measures for Mechatronics Systems, Verified and Supported by New MATLAB Built-in Function

Farhan A. Salem

Abstract

Mechatronics systems are supposed to operate with exceptional high levels of accuracy and speed despite adverse effects of system nonlinearities and uncertainties. Precise analytical expressions, as well as numerical codes for evaluating and verifying the achieving of desired performance and ensuring that all design requirements are met, are of concern and highly required. This paper proposes derivation of accurate analytical expressions for mechatronics systems performance measures with minimum possible deviation at actual values, that can be used to calculate, evaluate and verify the actual mechatronics systems' performance. To verify the accuracy and correctness of derived analytical expressions, as well as to calculate and return the precise numerical values of performance specifications, a new MATLAB built-in function is designed and used along with MATLAB software capabilities, where the actual normalized values are plotted and analyzed. The derived analytical expressions and new built-in function are only applied for first and second-order systems with no zeros and represent the essential qualities of higher-order systems with one or two dominant poles.

Keywords: Mechatronics systems, Response, Performance measures, Analytical expression, Built-in function

1. Introduction

Mechatronics is defined as multidisciplinary concept, it is synergistic integration of precision engineering mechanical engineering, electric engineering, electronic systems, information technology, intelligent control system, and computer hardware and software to manage complexity, uncertainty, and communication through the design and manufacture of products and processes from the very start of the design process, thus enabling complex decision making, exceptional levels of accuracy and speed of high-tech equipment including ability to perform complicated and precise movements of high quality, in result Mechatronics systems are to operate with high accuracy and speed despite adverse effects of system nonlinearities and uncertainties, therefore achieving and verifying accuracy in Mechatronics systems' performance is of concern, and the need for precise analytical expressions and numerical codes for calculating and verifying the achieving of desired accuracy of mechatronics systems performance is highly desired.

Most used formulae and expressions for performance specifications in texts including; (Robert H. Bishop, 2006)( Devdas Shetty et al, 1997)( De Silva et al, 2005) (Godfrey C. Onwubolu, 2005) (Ashish Tewari, 2002) (Katsuhiko Ogata, 1997)( Farid Golnaraghi et al, 2010) (Norman S. Nise, 2011)( Gene F. Franklin et al, 2002) (Bill Goodwine, 2011)( Dale E. Seborg et al, 2004)( Farhan A. Salem, 2013) lack accuracy, the determined performance specifications using these expressions, rarely accurate compared with actual results and measurements since it is more difficult to determine the exact analytical expressions of most used specifications and most introduced expressions are rough approximation of actual values. (Farhan A. Salem, 2013) precise analytical expressions for desired mechatronics time domain performance specifications, are derived and introduced, this paper proposes the derivation of more accurate expressions with minimum possible deviation at actual values, that can be used to calculate, evaluate and verify the actual mechatronics systems' performance, also this paper proposes a supporting new MATLAB built-in function for response plotting and performance measures precise calculating.

The performance of systems, the form and properties of response, are determined by the locations of its poles in Laplace domain. Many times, it is possible to identify a single pole, or a pair of poles, as the dominant poles. In such cases, a fair idea of the control system's performance can be obtained from the damping ratio $\zeta$ and undamped natural frequency $\omega_n$ of the dominant poles (Robert H. Bishop, 2006). Based on this, a designer can often make a
linear approximation to a nonlinear system. Linear approximations simplify the analysis and design of a system and are used as long as the results yield enough fair approximation to reality (Godfrey C. Onwubolu, 2005). A typical step response and its associated performance specifications of second order systems are shown in Figure 10(b). The most used time domain performance measures include; Time constant T, Rise time Tp, Settling time Ts, Peak time, Tp, Maximum overshoot Mp, maximum undershoot Mm, Percent overshoot OS%, Delay time Td. The decay ratio Dr, Damping period To and frequency of any oscillations in the response, the swiftness of the response and the steady state error e, (Farhan A. Salem, 2013)

2. Performance specifications of first order system.

First order systems without zeros and systems that can be approximated as first order systems, is characterized by performance measures, shown in Figure 1, including; time constant $T$, rise time $T_R$, settling time $T_s$, steady state error $e_s$, and DC Gain. When first order system is subjected to unit step, $R(s) = 1/s$, the response for these systems is natural decay or growth generated by the system single pole, the Laplace transform and solution of the step response are given by Eq. (1).

$$C(s) = R(s) \cdot C(s) = \frac{a}{s(s+a)} \quad \Leftrightarrow \quad c(s) = 1 - e^{-at}$$

$$c(s)|_{s=0} = 1 - e^{-\frac{1}{\alpha}} \quad \Rightarrow \quad 1 - e^{-1} = 0.63212055 \quad \text{Natural decay}$$

$$c(s)|_{s=\alpha} = e^{-\frac{1}{\alpha}} \quad \Rightarrow \quad e^{-1} = 0.36787944 \quad \text{Natural growth}$$

(1)

The introduced by (Farhan A. Salem, 2013) expressions for first order system performance measures are accurate enough for evaluating and verifying accurate mechatronics system performance, the only parameter needed to describe the transient response of a given first order system is system pole, $\alpha$.

2.1 Time constant $T$: value of time that makes the exponent of $e$ equal $-1$ is called the time constant $T$, it is a characteristic time that is used as a measure of speed of response to a step input and governs the approach to a steady-state value after a long time, the larger the time constant is, the slower the system response is. The time constant is the main characteristic unit of a first-order system, were the system reaches 63.21% of natural growth or 36.78% natural decay of its final value after time equals to one time constant, and reaches 99.3% of its final value after time equal to five time constants, Time constant $T$, is given by:

$$e^{at} = e^{-1} \Rightarrow at = -1 \Rightarrow T = \frac{1}{\alpha}, \quad \text{second}$$

(2)

Where: $\alpha$ is system pole. Time constant of first order system, can be found using any of the following approaches; using Eq.(2) or time for response $c(t)$ given by Eq.(1) to reach 63.2% (or 36.8% for exponential decrease) of its final value, $(1 - e^{-at}) = 1 - e^{-1/\alpha} = 0.632\%$ or system reaches 99.32% of its final value after time equals to 5T, geometrically the tangent drawn to the curve $e^{-at}$ at $t = 0$ intersects the time axis at the value of time equal to the one time constant $T$. The slope of the tangent line at $t=0$ is given by: $\text{slope}=1/T$, and the pole location in the s plane is given by: $\alpha = -1/T$.

2.2 Settling time $T_s$ is defined as the time required for the system output to fall within a particular certain percentage (1%, 2%, 5% or 10%) of the steady state value for a step input, the tighter the tolerance percentage, the longer the system response takes to settle to within this tolerance.

For 2% criterion, $T_s$ is found by equating Eq. (1) with 0.98, that is 1- 0.02 = 0.98 and solving for time, $t$, in terms of system pole, result in Eq.(3), or in terms of time constant as shown by Eq. (4).

$$0.98 = 1-e^{-at} \quad \Rightarrow \quad \ln 0.02 = \ln e^{-at} \quad \Rightarrow \quad \ln 0.02 = -\alpha \ln e \quad \Rightarrow \quad t = -\frac{\ln 0.02}{\alpha} \quad = \frac{3.9120}{\alpha} \quad (3)$$

$$0.98 = 1-e^{-T_s} \quad \Rightarrow \quad T_s = -\ln 0.02 = 3.9120/T$$

(4)

For 5% criterion, settling time $T_s$, is found by equating Eq. (1) with 0.95, that is 1- 0.05 = 0.95 and solving for time, $t$.

$$0.95 = 1-e^{-at} \quad \Rightarrow \quad \ln 0.05 = \ln e^{-at} \quad \Rightarrow \quad \ln 0.05 = -\alpha \ln e \quad \Rightarrow \quad t = -\frac{\ln 0.05}{\alpha} = \frac{2.9957}{\alpha} = 2.9957T \quad \text{Second}$$

2.3 Rise time $T_R$: is defined as the time required for the response curve to reach from particular criterion of lower level x% to some particular higher level y% of the final steady-state value, criterion levels’ ranges are from 10% to 90%, or 5% to 95% or 0% to 100% . Rise time is found by solving Eq. (1) for the difference in time.

2.4 Rise time for 10% to 90% criterion is found by solving Eq. (1) for the difference in time at $c(t) = 0.9$ and $c(t) = 0.1$, as shown and given by Eq. (5) or in terms of time constant as shown and given by Eq. (5). Rise time for other criterions are given in Eq(6)

$$T_{R} = \frac{1}{\alpha} \left(1 - e^{-\frac{1}{\alpha}}\right) = \frac{1}{\alpha} \left(1 - \frac{1}{e^{\frac{1}{\alpha}}}\right) = \frac{2.3026}{\alpha} - \frac{0.1054}{\alpha} = \frac{2.1972}{\alpha}$$

$$T_{R} = \frac{1}{\alpha} \left(1 - e^{-\frac{1}{\alpha}}\right) = \frac{1}{\alpha} \left(1 - \frac{1}{e^{\frac{1}{\alpha}}}\right) = \frac{2.3026}{\alpha} - \frac{0.1054}{\alpha} = \frac{2.1972}{\alpha}$$

$$T_{R} = \frac{1}{\alpha} \left(1 - e^{-\frac{1}{\alpha}}\right) = \frac{1}{\alpha} \left(1 - \frac{1}{e^{\frac{1}{\alpha}}}\right) = \frac{2.3026}{\alpha} - \frac{0.1054}{\alpha} = \frac{2.1972}{\alpha}$$

(5)

$$T_{R} = \frac{1}{\alpha} \left(1 - e^{-\frac{1}{\alpha}}\right) = \frac{1}{\alpha} \left(1 - \frac{1}{e^{\frac{1}{\alpha}}}\right) = \frac{2.3026}{\alpha} - \frac{0.1054}{\alpha} = \frac{2.1972}{\alpha}$$

(6)
2.5 The DC Gain: is defined as the ratio of the magnitude of the steady-state step response to the magnitude of the step input, and is given by Eq.(7), the DC gain indicates how much input voltage is required to reach a desired output value.

\[
 DCgain = \lim_{s \to 0} [sG(s).R(s)] = \lim_{s \to 0} [sG(s).\frac{1}{s}] = \lim_{s \to 0} [G(s)]
\]  

(7)

2.6 The steady-state error \( e_s \), is defined as the difference between reference input \( r(t) \) and actual final output \( c(t) \). It is the error after the transient response has decayed leaving only the steady state response, and is given by Eq. (8)

\[
 e_s = e(\infty) = E_s(s) = \lim_{s \to 0} \frac{sR(s)}{1 + G(s)} = \frac{1}{1 + \lim_{s \to 0} G(s)}
\]  

(8)

For underdamped case: \( 0 < \zeta < 1 \) and two complex conjugate poles given by \( -\zeta \omega_n \pm j \omega_n \sqrt{1-\zeta^2} \), general form of second order system can have the following form:

\[
 \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{s^2 + 2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}
\]

(9)

To obtain inverse Laplace transform we need to expand by partial fractions and solve, this all gives:

\[
 C(s) = \frac{1}{s} \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} + \frac{1}{s} \frac{s + 2\zeta\omega_n}{s^2 + 2\zeta\omega_n s + \omega_n^2}
\]

(10)

This can be rewritten to have the following forms:

\[
 c(t) = 1 - e^{-\zeta \omega_n t} \cos \omega_n t + \frac{\zeta}{\sqrt{1-\zeta^2}} \sin \omega_n t
\]

(11)

Eq. (10),(11)and(12) show that the damped natural frequency \( \omega_n \) given by \( \omega_n = \omega_n \sqrt{1-\zeta^2} \), is the frequency at which the system will oscillate if the damping is decreased to zero. The error signal for this system is the difference between input \( r(t) \) and output \( c(t) \) and is given by:

\[
 e(t) = e^{-\zeta \omega_n t} (\cos \omega_n t + \frac{\zeta}{\sqrt{1-\zeta^2}} \sin \omega_n t)
\]

(13)

3. Performance specifications of second order system

For second order systems, and systems that can be approximated as second order systems, there are four cases of response to consider; undamped, underdamped, critically damped and overdamped response. We shall solve for response performance specifications considering these cases, when system is subjected to unit step input, \( R(s) = 1/s \),

3.1 For underdamped case: \( 0 < \zeta < 1 \) and two complex conjugate poles given by \( -\zeta \omega_n \pm j \omega_n \sqrt{1-\zeta^2} \), general form of second order system can have the following form:

\[
 \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{s^2 + 2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}
\]

(11)

This can be rewritten to have the following forms:

\[
 c(t) = 1 - e^{-\zeta \omega_n t} \cos \omega_n t + \frac{\zeta}{\sqrt{1-\zeta^2}} \sin \omega_n t
\]

(12)

Eq. (10),(11)and(12) show that the damped natural frequency \( \omega_n \) given by \( \omega_n = \omega_n \sqrt{1-\zeta^2} \),is the frequency at which the system will oscillate if the damping is decreased to zero. The error signal for this system is the difference between input \( r(t) \) and output \( c(t) \) and is given by:

\[
 e(t) = e^{-\zeta \omega_n t} (\cos \omega_n t + \frac{\zeta}{\sqrt{1-\zeta^2}} \sin \omega_n t)
\]

(13)
\[ C(s) = \frac{1}{s^2 + \omega_n^2} \Rightarrow c(t) = 1 - e^{-\omega_n t} (1 + \omega_n t) \] (15)

3.4 For overdamped case; \( \zeta > 1 \), two distinct real poles, given by \( -\zeta \omega_n \pm j \omega_n \sqrt{\zeta^2 - 1} \) allow us to rewrite Eq.(9) to have the following form, when subjected to step input \( R(s) \):

\[ \frac{1}{s^2 + 2\zeta \omega_n s + \omega_n^2} = \frac{1}{s^2 + 2\zeta \omega_n^2 + \omega_n^2 \sqrt{\zeta^2 - 1}}(s + \frac{\omega_n}{\zeta \omega_n^2 - \omega_n^2 \sqrt{\zeta^2 - 1}}) \]

Taking inverse Laplace transform gives:

\[ c(t) = 1 + \frac{\omega_n}{2 \sqrt{\zeta^2 - 1}} \left( e^{-\alpha_1 t} - e^{-\alpha_2 t} \right) \] (16)

Where: \( \alpha_1, \alpha_2 \) are system poles. When \( \zeta \) is much greater than unity, i.e. \( \zeta >> 1 \), this response can be approximated, by neglecting one of systems two poles, the pole that is farther from imaginary axis, to have the unit-step time response of the following form:

\[ c(t) = 1 - e^{-\frac{\omega_n}{\sqrt{\zeta^2 - 1}} t} \] (17)

3.5. Deriving analytical expressions for performance specifications of underdamped second order system.

It is more difficult to determine the exact analytical expressions of the, rise time \( T_r \), settling time \( T_s \), and delay time \( T_d \). In (Farhan A. Salem, 2013), by applying mathematical concepts, curve fittings and trial and error approaches to Eq.(10) a precise analytical expressions, with minimum possible deviation at actual values were introduced, in this paper, applying similar concepts, more accurate analytical expressions are to be derived, particularly, for rise time \( T_r \), settling time \( T_s \), and delay time \( T_d \) are to be proposed, were other performance specifications and their corresponding analytical expressions derived and proposed by (Farhan A. Salem, 2013) are accurate enough to give accurate and with minimum possible deviation at actual values measure of systems' performance.

The correctness of derived analytical expressions for rise time \( T_r \), settling time \( T_s \), and delay time \( T_d \) as well as the correctness of used expression in different texts including (Robert H. Bishop, 2006), Devdas Shetty et al, 1997)( De Silva et al, 2005)( Godfrey C. Onwubolu, 2005)( Ashish Tewari, 2002)( Katsuhiro Ogata, 1997)( Farid Golnaraghi et al, 2010)( Norman S. Nise, 2011)( Gene F. Franklin et al, 2002)( Dale E. Seborg et al, 2004)( Farhan A. Salem, 2013), are to be compared, analyzed and verified, using MATLAB, were a specially designed MATLAB m.file and new built-in function were used, to determine performance specifications for given values of damping ratio over the range \( 0 \leq \zeta \leq 1.5 \), plots normalized times versus given \( \zeta \) for derived as well as most used specifications formulae and expressions in reference texts. The derived analytical expressions are only accurate for second-order systems with no zeros and represent the essential qualities of higher-order systems with two dominant poles. The definitions for settling time and rise
time are basically the same as the definitions for the first-order response. All definitions are also valid for systems of order higher than second.

3.5.1 Time constant T: The definitions of second order system time constant basically the same as the definitions for the first-order system. When the poles are complex quantity, \((\sigma \pm j\omega)\), the transient has the form of damped sinusoid \([ Ae^{\sigma t} \sin(\omega t + \phi)]\), in the case the time constant is defined in terms of complex pole real part \(\sigma\), that characterizes the envelope \( Ae^{\sigma t}\), see Figure 3(b), the time constant is equal to:

\[
T = 1/|\sigma| = 1/\zeta\omega_n
\]  

(18)

This equation shows that, the larger the product of \(\zeta\omega_n\), the greater the instantaneous rate of decay of the transient response.

3.5.2 Response maxima and minima.

3.5.2.1 Maximum overshoot \(M_p\) and minimum undershoot \(M_u\): Maximum overshoot \(M_p\) can be defined as the maximum value of the output, or it is the magnitude of the overshoot after the first crossing of the steady-state value, or the amount by which the system output response proceeds beyond the desired steady-state value. When output is lower than the final value, the phenomenon is called undershoot \(M_u\). Maximum overshoot = \(c_{max}(t) - c_o(\infty)\)

3.5.2.2 Maxima time or Peak Time \(T_p\): Peak time \(T_p\) is defined as the time required to reach the first maximum peak of the overshoot, it is a criterion of speed of response. The peak time is determined by finding the time when the derivative of Eq.(10) is equal to zero:

\[
\frac{dc}{dt}(t) = \zeta\omega_n e^{-\zeta\omega_n t}(\cos \omega_n t + \frac{\zeta}{\sqrt{1-\zeta^2}} \sin \omega_n t) - e^{-\zeta\omega_n t}(\sin \omega_n t + \frac{\zeta\omega_n}{\sqrt{1-\zeta^2}} \cos \omega_n t) = 0
\]

(19)

The values, that makes \(\frac{dc}{dt}(t) / dt = 0\), are \(0, \pi, 2\pi, 3\pi, 4\pi, \ldots = \pi\)

\[
\frac{dc}{dt}(t) = \sin \omega_n \sqrt{1-\zeta^2} t = 0 \Rightarrow \omega_n \sqrt{1-\zeta^2} t = n\pi
\]

The overshoots and undershoots occur at periodic intervals, this is shown in Figure 2, the overshoots occur at odd values of \(n\), the undershoots occur at even values of \(n\). The first overshoot is the maximum overshoot, this corresponds to \(n = 1\), the time at which the maximum overshoot occurs is peak time \(T_p\), and is derived by further solving, to have the following form:

\[
\sin \omega_n \sqrt{1-\zeta^2} t = 0 \Rightarrow \omega_n \sqrt{1-\zeta^2} t = n\pi \Rightarrow
\]

\[
t = T_p = \frac{n\pi}{\omega_n \sqrt{1-\zeta^2}}
\]  

(20)

Since the peak time \(T_p\), corresponds to the first peak overshoot (\(n = 1\)), we have:

\[
T_p = \frac{\pi}{\omega_n \sqrt{1-\zeta^2}} = \frac{\pi}{\omega_d}
\]  

(21)

Eq.(19) shows that the time, \(T_p\), is a function of both \(\zeta\) and \(\omega_n\). The exact magnitudes of the overshoots and the undershoots can be determined by substituting peak time given by Eq. (21) into Eq. (19), this gives the following:

\[
c_{max or min}(t) = M_p, or, M_u, = 1 - e^{-\zeta\pi \sqrt{1-\zeta^2}} \sin(n\pi + \phi)
\]

\[
M_p, or, M_u, = 1 - (-1)^n e^{-\zeta\pi \sqrt{1-\zeta^2}} \sin(n\pi + \phi)
\]  

(22)

The maximum overshoot is obtained by letting \(n=1\) in Eq. (22), this gives:

\[
M_p = e^{-\zeta\pi \sqrt{1-\zeta^2}} \Rightarrow M_p = \exp(-\frac{\zeta\pi}{\sqrt{1-\zeta^2}})
\]  

(23)

This equation shows that, the maximum overshoot \(M_p\) is a function of only the damping ratio \(\zeta\), therefore it is used to evaluate the damping of the system, smoothness of response.

3.5.2.3 Percent maximum overshoot \(\%OS\): Percent overshoot is the amount that the response overshoots the steady state final value at the peak time, expressed as a percentage of the steady-state value, and can be derived as follows:

\[
\text{OS} \% = 100 \times \frac{\left( c(t_p) - c(\infty) \right)}{c(\infty)} = 100 \times \frac{e^{-\zeta\pi \sqrt{1-\zeta^2}} - 1}{1}
\]

\[
\text{OS} \% = 100 \times e^{-\zeta\pi \sqrt{1-\zeta^2}}
\]  

(24)

Percent maximum overshoot \(\%OS\) measures the closeness of the response to the desired response; also it is a relative stability criterion, with 10% to 20% as an acceptable value. It is good to increase damping ratio, the decrease in the damping ratio \(\zeta\), leads to increase the overshoot and the time response. For given percent maximum overshoot \(\%OS\), the damping ratio can be found by rearranging Eq. (24) to have the following form:

\[
\zeta = \frac{-\ln(\%OS/100)}{\sqrt{\%OS/100} + \ln(\%OS/100)}
\]  

(25)

For desired \(\%OS\) less than 18% then \(\zeta \geq 0.479\), and if 10% of \(\%OS\) is acceptable then \(\zeta \geq 0.591\).

3.5.2.4 Minima Time \(T_U\): The first undershoot is the minimum undershoot, this corresponds to \(n=2\), in Eq. (21), the time at which the minimum undershoot occurs is Minima Time \(T_M\), and is given by further solving, to have the following form

\[
T_U = \frac{2\pi}{\omega_n \sqrt{1-\zeta^2}} = \frac{2\pi}{\omega_d}
\]

3.5.2.5 The decay ratio \(D_h\): Is defined as the exponential decay between successive peaks; the first maximum
overshoot and the second peak overshoot. It is the ratio of the second overshoot divided by the first, with 1/4 a common design value; this specification is often used in process control industry. Referring to Eq. (21), the second upper overshoot $M_{p2}$ occurs at value $n=3$, the decay ratio is given by:

$$D_n = \frac{M_p}{M_{p2}} = \frac{e^{-2\pi \sqrt{1-\xi^2}}}{e^{-3\pi \sqrt{1-\xi^2}}} = e^{-\pi \sqrt{1-\xi^2}}$$

(26)

The decaying ratio and maximum overshoot are functions of damping ratio only.

3.5.2.6 The damping factor $D_\xi$: Is defined as the ratio of first maximum overshoot divided by the first undershoot, the maximum overshoot is obtained by substituting $n=1$ in Eq. (21), the first undershoot is obtained by substituting $n=2$, therefore the expression for damping factor is given by:

$$D_\xi = \frac{e^{-\pi \sqrt{1-\xi^2}}}{e^{-2\pi \sqrt{1-\xi^2}}} = e^{-\pi \sqrt{1-\xi^2}}$$

(27)

3.5.2.7 The period, $T_o$ and frequency of any oscillations in the response: The period of oscillation is the amount of time between two successive upper overshoot peaks; its formula can be derived by subtracting time of first and second overshoots peaks. The period of oscillation is given by:

$$T_o = T_{p2} - T_p = \frac{3\pi}{\omega_0 \sqrt{1-\xi^2}} - \frac{\pi}{\omega_0 \sqrt{1-\xi^2}} = \frac{2\pi}{\omega_0 \sqrt{1-\xi^2}}$$

(28)

The following relationship exists between, The peak Time, $T_p$ given by Eq.( 21) and the oscillation period, were the peak time is half the oscillation period, as shown next:

$$T_p = 0.5T_o = \frac{1}{2} \frac{2\pi}{\omega_0 \sqrt{1-\xi^2}} = \frac{\pi}{\omega_0 \sqrt{1-\xi^2}} = \frac{\pi}{\omega_0}$$

The frequency of oscillation is given by:

$$\omega = \omega_0 = \frac{1}{T} = \frac{\omega_0 \sqrt{1-\xi^2}}{2\pi}$$

(29)

3.5.2.8 Settling time $T_s$: for step input, $T_s$ is defined as the time required for the response curve to reach and stay within a range about the final value of size specified by absolute percentage of final value, usually 2% or 5% (Devdas Shetty et al., 1997). In other words, Settling time is the time the response curve takes to meet its desired final state value, $T_s$ is a criterion of both of speed of response and stability, it reflects both response speed and damping. Settling time should be as small as possible because smaller values represent a faster response and an ability to reduce costs. For 2% criterion, the settling time is the time for which response $c(t)$ in Eq. (10) reaches and stays within band of ±2% of the final steady-state value:

$$c(t) = 1 - \frac{1}{\sqrt{1-\xi^2}} e^{-\omega_0 \sqrt{1-\xi^2} t} \cos(\omega_0 t - \phi) = 0.02$$

The transient response decays is governed by the exponential term $e^{-\omega_0 x}$, assuming $\cos(\omega_0 t - \phi) = 1$, at the settling, and solving we have:

$$T_s = -\ln(0.02) \frac{\sqrt{1-\xi^2}}{\xi \omega_0}$$

(26)

From this equation, the settling time for tolerance band of x per cent, may be obtained using the following expression:

$$T_s = -\ln\left(\frac{x}{100} \frac{\sqrt{1-\xi^2}}{\xi \omega_0}\right)$$

(27)

Plotting $T_s$ versus numerator of Eq.(26) with varying values of damping $\zeta = 0:0.01:1$, (see Figure 3(a)) shows that the numerator will vary from 3.912 to 5.81, the corresponding to $\zeta$ numerator value can then be divided over $\xi \omega_0$ to obtain the value of settling time. analytical expression for settling time can be obtained by analyzing and comparing both actual values of settling time obtained from Figure 3(a), and results of running the below MATLAB code with defined tf and zetas, this code will return the actual values of settling time for given range of damping ratio $0 \leq \zeta \leq 1.5$, as well as plot the actual settling time versus various values of damping ratio.

$$\text{actual_Ts}=[];$$
$$Q=(abs(y(:,i))>0.05); \% For 2\% criterion for i=1:length(zeta)$$
$$\text{AA}=[Q(:)];$$
$$\text{actual_Ts(i)=[A(length(AA))];}$$
$$\text{end}$$
$$\text{actual_Ts;}$$
$$\text{plot(zeta,10*actual_Ts,'r'),hold on}$$

The plot and results show that, for $0 \leq \zeta < 0.7$, the unit-step response has a maximum overshoot greater than 5%, and the response can enter the band between 0.95 and 1.05 for the last time from either the top or the bottom. When $\zeta$ is greater than 0.7, the overshoot is less than 5%, and the response can enter the band between 0.95 and 1.05 only from the bottom. It is difficult to obtain exact analytical expression of the settling time $T_s$, but using the envelope of the damped sinusoid shown in Figure 3(b), it is possible to obtain an approximation for $T_s$ for $0 < \zeta < 0.7$, where when the settling time corresponds to an intersection with the upper or the bottom envelope of $c(t)$, the following corresponding relation are obtained:

$$1 - \frac{e^{-\omega_0 T_s}}{\sqrt{1-\xi^2}} = \text{Upper} \iff 1 - \frac{e^{-\omega_0 T_s}}{\sqrt{1-\xi^2}} = \text{Lower} \iff 0 < \zeta < 0.7$$

The same result for $T_s$ is obtained using either equation, solving for 5% $T_s$, gives:

$$T_s = \frac{1}{\xi} \ln(0.05) \frac{\sqrt{1-\xi^2}}{\xi \omega_0}$$

(30)

Plotting $T_s$ versus numerator with varying values of $\zeta = 0:0.1:1$ shows that the numerator will vary from 3 to 4.951, but as $\zeta$ varies from 0 to 0.7, $T_s$ varies between 3.0 and 3.32. To obtain the settling time, the corresponding to $\zeta$ numerator value can then be divided over corresponding
\( \zeta \). The following suggested simplified approach, can give an approximate expression for settling time with minimum deviations from actual values; since the rate at which the transient response decays is governed by the exponential term \( e^{-\zeta \omega_n t} \) the settling time for the 2% and 5% criterion can be calculated by:

\[
e^{-\zeta \omega_n t} = 0.02 \Rightarrow T_S = \frac{\ln(0.02)}{\zeta \omega_n} = 3.912 \quad \Rightarrow \ 0 < \zeta < 0.7
\]

\[
e^{-\zeta \omega_n t} = 0.05 \Rightarrow T_S = \frac{\ln(0.05)}{\zeta \omega_n} = 2.9957 \quad \Rightarrow \ 0.7 < \zeta < 1.5
\]

(28)

3.5.2.9 Settling time \( T_s \), 5% criterion,

For \( \zeta > 0.7 \) the value of \( T_S \) is almost directly proportional to \( \zeta \), through curve analyzing, curve fitting and trial and error, the following approximation can be suggested:

\[
T_S = 4.59 \frac{\zeta}{\omega_n} \quad \Rightarrow \quad \zeta > 0.7
\]

More better and more accurate expressions for \( \zeta > 0.7 \) was chosen through curve analyzing, curve fitting and trial and error, and given by:

\[
T_S = \frac{(6.68 \zeta - 1.9084)}{\omega_n} \quad \Rightarrow \quad 0.7 \leq \zeta \leq 1.2
\]

(29)

\[
T_S = \frac{(6.57 \zeta - 1.64)}{\omega_n} \quad \Rightarrow \quad 1.2 < \zeta < 1.5
\]

(30)

Different formulae and different approximations for the settling time appear in different texts, Referring to (Farid Golnaraghi et al, 2010) settling time is given by:

\[
T_S = \frac{3.2}{\zeta \omega_n} \quad \Rightarrow \quad 0 < \zeta < 0.7
\]

\[
T_S = \frac{4.5 \zeta}{\omega_n} \quad \Rightarrow \quad \zeta > 0.7
\]

(31)

Referring to (Gene F. Franklin et al, 2002) settling time is given by:

\[
T_S = \frac{4.6}{\zeta \omega_n} \quad \Rightarrow \quad 0 < \zeta < 0.7
\]

(32)

Referring to (w.courses.engr.illinois.edu) settling time is given by:

\[
T_S = \frac{-0.5 \ln((1 - \zeta^2)/400)}{\zeta \omega_n} \quad \Rightarrow \quad \zeta < 0.7
\]

\[
T_S = \frac{(6.6 \zeta - 1.6)}{\omega_n} \quad \Rightarrow \quad \zeta > 0.7
\]

(33)

Figure 3(a) \( T_s \) versus \(-\ln(0.02\sqrt{1-\zeta^2})\) with \( \zeta = 0:0.1:1 \).

To verify the correctness of derived expressions for settling time with the actual settling time values and to compare the results of derived expressions with different expressions appear in different texts, we plot normalized time \( \omega_n T_R \) versus \( \zeta \), of all expressions in the same graph window and compare it with actual settling time values.

The resulted plots are shown in Figure 4(a), these plots show that for \( \zeta > 0.7 \) the actual settling times rise smoothly as \( \zeta \) increases, which results in slowing down the system response. For this range the suggested two expressions given by Eqs.(29) and (30) are most accurate expressions that all mostly fit the actual settling time with average difference of 0.07 seconds. For \( 0.7 < \zeta < 1.2 \) the suggested expression given by Eq.(30) almost fit the actual values of settling time with average difference of 0.03 seconds. For \( 0.1 > \zeta > 0.7 \), actual settling time peaks at times equal to, 0.12, 0.14, 0.16, 0.19, 0.24, 0.31, 0.44 and at 0.69, for the peak between 0.44 and at 0.69 the following expression can be suggested:

\[
T_S = \frac{5.246}{\omega_n} \quad \Rightarrow \quad 0.44 < \zeta < 0.69
\]

(34)

For the peak between \( 0.0 > \zeta > 0.31 \), the following expression can be suggested:

\[
T_S = \frac{2.88}{\omega_n} \quad \Rightarrow \quad 0 < \zeta < 0.301
\]

(35)

It would be difficult to model each peak exactly; the expression given by Eq.(28) can be used to calculate the approximate values of settling time as well as maximum upper limit for settling time at this range. A suggested expression for the lower limit of settling time is given by:

\[
T_S = \frac{2.5057}{\zeta \omega_n} \quad \Rightarrow \quad 0 < \zeta < 0.301
\]

(36)

The average difference in time between the upper and lower limits is about ± 0.025, based on this, following average expression can be suggested for \( 0.1 > \zeta > 0.31 \):

\[
T_S = \frac{7.8}{\omega_n} \quad \Rightarrow \quad 0.301 < \zeta < 0.44
\]
In Figure 4(b) both, actual settling time values and values calculated using derived expressions are plotted, these curves show that the suggested analytical expression can be used to calculate the settling time (5% criterion) for second order systems, and systems that can be approximated as second order with minimum deviation at actual values.

\[ T_s = \frac{2.7507}{\xi \omega_n} \Rightarrow 0 < \xi < 0.301 \]  \hspace{1cm} (37)

For the peak between 0.53 and at 0.78 the following expression can be suggested:

\[ T_s = \frac{5.8}{\omega_n} \Rightarrow 0.53 < \xi < 0.78 \]  \hspace{1cm} (38)

For the peak between 0.78 and 1.2 at the following expression can be suggested:

\[ T_s = (9.309 -3.53) \frac{\xi}{\omega_n} \Rightarrow 0.78 < \xi < 1.2 \]

In Figure 5 both, actual settling time values and values calculated using derived expressions are plotted, these curves show that the suggested analytical expression can be used to calculate the settling time (2% criterion) for second order systems, and systems that can be approximated as second order with minimum deviation at actual values.

Running the designed m.file and built-in function for different values and calculating the percentage error, it was found that it is in the range of 0.01-0.04, for different damping ratio ranges, which reflect the precision of derived expression and their applicability in accurate evaluation of system performance. Running the supporting new MATLAB built in function for damping ratio of 0.6 and undamped natural frequency of 2, will give the below results, and response curve shown in figure 6:

| Actual settling time, 0.02 criterion, Ts = 2.96706 Seconds |
| Calculated settling time, 0.02 criterion, Ts = 2.96 Seconds |
| Actual settling time, 0.05 criterion, Ts = 2.59618 Seconds |
| Calculated settling time, 0.05 criterion, Ts = 2.623 Seconds |

---

**Figure 4(a)** Comparing different analytical expressions for TR with actual values.

**Figure 4(b)** Curves of settling time both actual and calculated by derived expression 5% criterion.

**3.5.2.10 Settling time Ts, 2% criterion.**

The same procedure was applied, to derive analytical expression, to be as follows:

- For the peaks between 0.0 > \( \xi > 0.39 \), the following expression can be suggested:
  \[ T_s = \frac{3.75}{\omega_n} \Rightarrow 0 < \xi < 0.39 \]

- For the peak between 0.39 and at 0.53 the following expression can be suggested:
  \[ T_s = \frac{8.4}{\omega_n} \Rightarrow 0.39 < \xi < 0.53 \]

Figure 5 Curves of settling time both actual and calculated by derived expression 2% criterion.
3.5.2.10 Rise time \( T_R \).

Rise time \( T_R \) is a measure of swiftness of response, it is the time required for the response curve to reach from 10% to 90%, or 5% to 95% of its final value for critical and overdamped cases, or 0% to 100% of its final value for underdamped cases. An alternative measure is to represent the rise time as the reciprocal of the slope of the step response at the instant that the response is equal to 50% of its final value (Farid Golnaraghi et al, 2010), that is at delay time \( T_D \). It is difficult to determine precise analytical expressions for rise time \( T_R \). Different approximate formulae for the rise time appear in different texts. One reason for the different formulae is because of different definitions of the rise time (Bill Goodwine, 2011), other reason could be the required accuracy. The exact values of rise time for given range of damping ratio, can be determined directly from the responses of Figure 2(a), or over the range 0 < \( \zeta < 1.5 \). We first designate \( \omega_n \) for that value of \( \zeta \), and solve for the values of \( \omega_n \), 0% to 100% criterion that yield \( T_R \) and \( T_D \). Normalized rise time \( \tau_R \) for ω = \( \omega_n \) is defined be the normalized time variable and select a value for \( \omega_n \).

These equations imply that rise time increases as damping increases. An approximation techniques can be used to estimate approximate values; by plotting normalized time \( \omega_n T_R \) versus range of 0 < \( \zeta < 1.5 \), and then approximate the curve by a straight line or over the range of 0 < \( \zeta < 1.5 \). We first designate \( \omega_n T_R \) as the normalized time variable and select a value for \( \zeta \).

For underdamped case; 0% to 100% of its final value, the rise time can be obtained by equating Eq.(10) with unity and solve for time \( t \), that is rise time:

\[
c(t) = 1 - e^{-\zeta \omega_n T_R} \left( \cos \omega_n T_R + \frac{\zeta}{\sqrt{1-\zeta^2}} \sin \omega_n T_R \right) = 1
\]

Since \( e^{-\zeta \omega_n T_R} \neq 0 \), we have:

\[
\cos \omega_n T_R + \frac{\zeta}{\sqrt{1-\zeta^2}} \sin \omega_n T_R = 0 \Rightarrow \tan \omega_n T_R = -\frac{\sqrt{1-\zeta^2}}{\zeta}
\]

\[
\tan \omega_n T_R = -\frac{\omega_n \sqrt{1-\zeta^2}}{\zeta \omega_n} \Rightarrow T_R = \frac{1}{\omega_n} \tan^{-1} \left( \frac{\omega_n}{\zeta} \right)
\]

\[
T_R = \frac{\pi - \phi}{\omega_n} = \frac{\pi}{2 \omega_n} \frac{\pi}{\omega_n \sqrt{1-\zeta^2} / \zeta} \rightarrow 0 < \zeta < 1
\]

Where referring to Figure 7 (a), \( \phi \) is defined by the following Eqs.:

\[
\phi = \cos^{-1} (\zeta) \Leftrightarrow \phi = \sin^{-1} \left( \frac{\sqrt{1-\zeta^2}}{\zeta} \right) \Rightarrow \phi = \tan^{-1} \left( \frac{\sqrt{1-\zeta^2}}{\zeta} \right)
\]

In the limit as \( \zeta \rightarrow 0 \), this equation can be approximated as:

\[
T_R = \frac{\pi - \pi / 2}{\omega_n} = \frac{\pi}{2 \omega_n}
\]

In the limit as \( \zeta \rightarrow 1 \), this equation can be approximated as:

\[
T_R = \frac{\pi - 0}{\omega_n \sqrt{1-\zeta^2}} = \frac{\pi}{\omega_n \sqrt{1-\zeta^2}}
\]

For underdamped case; 0% to 100% of its final value, the rise time can be obtained by equating Eq.(10) with unity and solve for time \( t \), that is rise time:

![Figure 7 (a) Definition of angle \( \phi \).](image-url)
Curve fitting can be applied to curve shown in Figure 7(b), to derive an approximate third order approximation given by:

\[
T_R = 1.765\xi^3 - 0.417\xi^2 + 1.039\xi + 1 \quad \Rightarrow 0 < \xi < 0.9
\]  

(39)

Quadratic approximation can result in the following expressions:

\[
T_R = 2.230\xi^2 - 0.078\xi^2 + 1.12 \quad \Rightarrow 0 < \xi < 0.9
\]  

(40)

Referring to (Farid Golnaraghi et al, 2010), the rise time \(T_r\) for second order underdamped system, rise time can be approximated as a straight line given by:

\[
T_R = \frac{0.8 + 2.5\xi}{\omega_n} \quad \Rightarrow 0 < \xi < 1
\]  

(41)

Referred to (Norman S. Nise, 2011), the linear approximation of rise time is given by:

\[
T_R = 0.6 + 2.16\xi \quad \Rightarrow 0.3 < \xi < 0.8
\]  

(42)

Referring to (Gene F. Franklin et al, 2002), rise time is given by:

\[
T_R = \frac{2.2}{\xi\omega_n}
\]  

(43)

Referring to (Farid Golnaraghi et al, 2010), rise time is given by:

\[
T_R = \frac{1.2 - 0.45\xi + 2.6\xi^2}{\omega_n} \quad \Rightarrow \xi < 1.2
\]

(44)

These equations show that rise time is proportional to \(\xi\) and inversely proportional to \(\omega_n\). Most of these derived expression are rough approximations and mostly has huge deviation at actual values, this is shown in Figure 7(c), analyzing actual curve, shown in Figure 7(b), shows that the curve can be fit as ramp in some regions and of second order in others, applying curve fitting and trial and error approaches, better and more accurate expressions, can be suggested for \(0 \leq \xi < 0.4\), for \(0.4 \leq \xi < 1.2\), and for \(\xi > 1.2\), suggested analytical expressions are given by:

\[
T_R = \frac{1.2 - 0.2\xi + 3\xi^2}{\omega_n} \quad \Rightarrow 0 < \xi < 0.4
\]

\[
T_R = \frac{1.26 - 0.51\xi + 2.58\xi^2}{\omega_n} \quad \Rightarrow 0.4 \leq \xi < 1.2
\]

\[
T_R = \frac{4.67\xi - 1.2}{\omega_n} \quad \Rightarrow \xi > 1.2
\]  

(45)

Plotting actual rise time and a rise time obtained using suggested expressions, both versus normalized time, are shown in Figure 7(d), analysis of both plots show that the suggested expressions match the actual values with maximum upper error of 0.06 seconds for \(0.34 < \xi < 0.4\), and maximum upper error of 0.026 seconds for \(0.6 < \xi < 0.7\). The suggested expressions can be used to analytically calculate rise time with error of ± 0.02 seconds.
Plotting actual rise time and a rise time obtained using suggested expressions, both versus normalizes time, are shown in Figure 7(e).

Figure 7(e), Rise time, 10%–90%, plot of actual and using derived expressions.

Running the designed m.file and built-in function for different damping ratio ranges values and calculating the percentage error, it was found that it is in the range of 0.002–0.012. Running the supporting new MATLAB built in function for damping ratio of 0.6 and undamped natural frequency of 2, will give the below results, and response curve shown in Figure 8:

Actual Rise time, 0.01-0.09 criterion, $T_{r} = 0.92803$ Seconds  
Calculated Rise time, 0.01-0.09 criterion, $T_{r} = 0.92940$ Seconds

Figure 8

3.5.2.12 Delay time $T_{D}$ (Half final value time, $T_{50}$):  
Also called 50% rise time, It is defined as the time required for the response to reach half of final value the very first time (Katsuhiko Ogata, 1997). Delay time can be determined directly from the responses of Figure 5(a). It is difficult to determine the exact analytical expressions of the delay time, an approximation technique can be used, to plot normalized time $\omega_{n}T_{D}$ versus $\zeta$, and then approximate the curve by a straight line or approximate the curve over the range of $0 < \zeta < 1$, another approach we can set Eq. (11) equal to 0.5 and solve for delay time $t$, also based on the definitions of the delay time $T_{D}$ and rise time $T_{r}$, an analytical expression for delay time can be derived, since the waveform between $0.1c(s)$ to $0.9c(s)$ is not a linear line, the following expression with rough approximation for delay time in terms of rise time, can be suggested:

$$T_{D} = 0.67 \cdot T_{r}$$  (47)

Softening this expression can be accomplished as follows, since response depends proportionally on damping ratio and inversely on undamped natural frequency, also Figure 9 shows that the delay time curve is of second order, adding these factors to Eq.(56), applying curve fittings and trial and error methods, the following expression for delay time in terms of damping ratio $\zeta$, undamped natural frequency $\omega_{n}$ and rise time $T_{r}$, can be suggested:

$$T_{D} = \frac{0.5 \cdot \omega_{n}^{2} T_{r} + 0.4 \zeta + 1.1}{\omega_{n}}$$  (48)

Applying the same approach, the following expression for delay time, in terms of damping ratio $\zeta$ and undamped natural frequency $\omega_{n}$, can be suggested:

$$T_{D} = \frac{(1.2378 - 0.156 \zeta + 0.592 \zeta^{2})}{\omega_{n}}$$  (49)

Referring to (Farid Golnaraghi et al, 2010), the delay time $T_{D}$, for second order underdamped system, can be approximated as a straight line given by:

$$T_{D} \equiv 1 + \frac{0.7 \zeta}{\omega_{n}} \rightarrow \cdots 0 < \zeta < 1$$  (50)

A better approximation by using a second-order approximation is given by:

$$T_{D} \equiv \frac{1.1 + 0.125 \zeta + 0.469 \zeta^{2}}{\omega_{n}} \rightarrow \cdots 0 < \zeta < 1$$  (51)

Equations show that delay time is proportional to $\zeta$ and inversely proportional $\omega_{n}$, a comparison between actual and analytical values of delay time, using different expressions are shown in Figure 9(a). A comparison between delay time values, actual and obtained analytically obtained using suggested expressions given by Eqs. (84) and (49), are shown in Figure 9(b), plots show that both suggested expressions can be used to calculate delay time with overage error of 0.005 seconds, also expression based on rise time and damping ratio given by Eqs. (48) gives more accurate results over given range of damping ratio, where expression given by Eqs. (49) gives results with increasing error for all $0 < \zeta < 0.39$. 

274
4. New MATLAB built-in function

The introduced new MATLAB built-in function, is named ‘mechspec’, it is short for mechatronics specifications, based on derived analytical expressions, and MATLAB matrix algebra capabilities, this new built-in function can be used to return, both accurate actual and precise calculated performance specifications, to be used to verify the correctness and accuracy of derived analytical expressions, this function is also can be used to calculated the precise numerical values of performance specification of a given first or second order system in terms of: Time constant, $T$, Rise time $T_r$, Settling time $T_s$, Peak time, $T_p$, Maximum overshoot $M_p$, maximum undershoot $M_u$, Percent overshoot $O(s)$, Decay time $T_d$, The Undamped natural frequency, $\Omega_{n}$, The Damped natural frequency, $\Omega_{d}$, The decay ratio $D$, Damping period $T_o$ and frequency of any oscillations in the response, the swiftness of the response and the steady state error $\epsilon_{ss}$, also this built-in function will plot step response, running this built in function for given second order system given by (52), will return the below performance specifications and step response curve shown in figure 10(a).

Figure 9(a). Comparison between actual and analytical delay time, using different expressions.

Please wait: it takes 3-5 seconds

num/den =

\[ \frac{6}{s^2 + 2s + 6} \] (52)

Time constant, $T$ Calculations:

Stability analysis:
The system poles, $\lambda 1 = -1.00, \lambda 2 = -1.00$
The system is stable

\[ \zeta = 0.40825 \text{ Sec.} \]

\[ \Omega_n = 2.44949 \text{ rad/s} \]

\[ \Omega_d = 2.23607 \text{ rad/s} \]

\[ T = 1.00000 \text{ Seconds} \]

System reaches 99.3 of its final value, after $5T = 5.00000$ Seconds

Actual DC gain, DC gain = 1.00000
Calculated DC gain2, DC gain = 1.00000
Steady state error, $\epsilon_{ss} = 0.50000$ Seconds

Delay time, $T_d$ Calculations:

Actual Delay time, $T_d = 0.5196$ Sec.

Rise time, $T_r$ Calculations:

Actual Rise time, (0.1-0.09 criterion), $T_r = 0.58837$ Seconds
Calculated Rise time, (0.1-0.09 criterion), $T_r = 0.59827$ Seconds

Actual Rise time, (0.0-1.0 criterion), $T_r = 0.60394$ Seconds
Calculated Rise time, (0.0-1.0 criterion), $T_r = 0.60494$ Seconds

Settling time, $T_s$ Calculations:

Actual settling time, 0.02 criterion, $T_s = 3.42959$ Seconds
Calculated settling time, 0.02 criterion, $T_s = 3.42929$ Seconds

Actual settling time, 0.05 criterion, $T_s = 3.18333$ Seconds
Calculated settling time, 0.05 criterion, $T_s = 3.18434$ Seconds

Percent Overshoot, $M_p$ Calculations:

Actual max. Percent Overshoot, $M_p = 24.52809$
Calculated max. Percent Overshoot, $M_p = 24.53761$
Calculated max. Undershoot, $M_u = 0.93979$

Peak time, $T_p$ Calculations:

---

**Calculated Peak time, $T_p = 1.40496$ Seconds**

---

Calculated MINIMA time, $T_u = 2.80993$ Sec
Calculated decay ratio, $D_R = 0.06021$
Calculated damping factor, $D_\zeta = 4.66021$
Calculated Period of oscillation, $T_0 = 2.80993$
Calculated Frequency of any oscillation in the response, $\Omega = 2.23607$ rad/sec

---

```matlab
function mecspec( num, den )
% mecspec A new built in function for returning accurate numerical values
% of performance specifications for applied , for first and second order systems and systems approximated as first or second order
clc, close all,
q=length(den);
sys1=tf(num, den);
if q==2
    root=roots(den); poles=abs(root);
    if poles < 0
        uu=' The system is stable';
    else
        uu=' The system is unstable';
    end
    Time_constant = -1/( poles);
    % rise time calculations, $T_r$ Calculations
    rise_time_1_9=2.1972/poles;
    rise_time_0_1=4.6/poles;
    % settling time calculations, $T_s$ Calculations
    calculated_098_Ts=3.9120/poles;
    % Steady state error, $E_s$ Calculations
    Ess= 1/(1+ dcgain(sys1));
    % DC gain, Calculations
    t=0:0.001:100;% time and increment
    y=step(sys1,t);
    DC_gain2=y(length(t));
    % Steady state error, $E_s$ Calculations
    Ess= 1/(1+ dcgain(sys1));
    home, printsys(num,den,'s')
    disp( '  '), sys1; step(sys1)
    disp(  '  ')
    disp('  ')
    disp('  ')
end
figure 10(a) Step response and specifications
Figure 10(b) Second-order underdamped response specifications

4.1 built-in function code:

function mecspec( num, den )
% mecspec A new built in function for returning accurate numerical values
% of performance specifications for applied , for first and second order systems and systems approximated as first or second order
```
disp('Actual Rise time,(0.1-0.09 criterion), Tr = \%2.5f Seconds \n'); actual_Tr_1_9 = %2.5f
\nfprintf('%s','Calculated Rise time,(0.1-0.09 criterion), Tr = \%2.5f Seconds 
'); rise_time_1_9 = %2.5f
\nfprintf('%s','Calculated Rise time,(0.0-1.0 criterion), Tr = \%2.5f Seconds 
'); rise_time_0_1 = %2.5f
\ndisp('Actual settling time,0.98 criterion, Ts = \%2.5f Seconds 
'); actual_Ts2 = %2.5f
\nfprintf('%s','Calculated settling time,(0.98 criterion), Ts = \%2.5f Seconds 
'); calculated_098_Ts = %2.5f
\ndisp('Actual settling time,0.98 criterion, Ts = \%2.5f Seconds 
'); actual_Ts1 = %2.5f
\nfprintf('%s','Calculated settling time,(0.98 criterion), Ts = \%2.5f Seconds 
'); calculated_098_Ts = %2.5f
\n[sys]=tf(omega_n*omega_n, [1, 2*zeta*omega_n, omega_n*omega_n]); % omega_n=1
\n[y,t]=step(sys{:});
\n% Calculating Actual settling time using MATLAB
thresh1=0.05; % 5 percent - bigger than in Franklin
sel1=logical(abs(y(:,:,i))>0.05);
\nactual_Ts1=[ ];
for i=1:length(zeta1)
tmp1=step(1:length(zeta1),ii);
actual_Ts1(i)=tmp1(length(tmp1));
end
\nactual_Ts1=actual_Ts1';
\na1=find(answer1(:,1)== zeta);
actual_Tr_1_9=answer1(a1,2);
\nthresh1=0.02; % 2 percent - bigger than in Franklin
sel2=logical(abs(y(:,:,i))>0.02);
\nactual_Ts2=[];
for i=1:length(zeta1)
tmp2=step(1:length(tmp2),ii);
actual_Ts2(i)=tmp2(length(tmp2));
end
\nactual_Ts2=actual_Ts2';
\na2=find(answer2(:,1)== zeta);
actual_Tr_1_9=answer2(a2,2);
\n% Calculating setti
...
elseif 0.4 <= zeta && (zeta < 0.85)
    Tr_calcul_1_9=(1.26 -0.55*zeta+2.58.*zeta^2)/omega_n;
elseif 0.85 <= zeta && (zeta <= 1.2)
    Tr_calcul_1_9=(1.20 -0.45*zeta+2.58.*zeta^2)/omega_n;
else
    Tr_calcul_1_9=(4.69*zeta -1.19)/omega_n;
end
% Calculating Rise time ,Tr ; 0%%:100%
A0=logical( y(:,:)>=0);
A1=logical( y(:,:)>=0.999);
time_0=[];
time_1=[];
Rise_time_0_1=[];
for i=1:length(zeta1)
    tmp0= t(A0(:,i));
    tmp1= t(A1(:,i));
    time_0(i)=tmp0(i);
    time_1(i)=tmp1(i);
end
Rise_time_0_1=time_1-time_0;
answer4=[ zeta1, Rise_time_0_1'];
a4=find(answer4(:,1)== zeta);
actual_Tr_0_1=answer4(a4,2);
if ((0< zeta)  && (0.4 > zeta))
    Tr_calcul_0_1 = (1.2 -0.2*zeta+3.*zeta^2)/omega_n;
elseif 0.4 <= zeta  && (zeta  <  1.2)
    Tr_calcul_0_1=(1.26 -0.51*zeta+2.58.*zeta^2)/omega_n;
else
    Tr_calcul_0_1=(4.67*zeta -1.2)/omega_n;
end
Tr_calcul_0_1;
% Percent maximum Overshoot,Mp Calculations
[y1,t1]=step([sys1]);
actual_Mp=100*(max(y1(:,:))-1);
% [y,t]=step(sys1);
Mp=max(y(:,:))-1;
% answer5=[zeta1, Mp'];
% a5=find(answer5(:,1)== zeta);
% actual_Tr=answer4(a4,2);
% if ((zeta > zeta)  && (0.4 > zeta))
%     Tr_calcul_0_1 = (1.2 -0.2*zeta+3.*zeta^2)/omega_n;
% elseif 0.4 <= zeta  && (zeta <= 1.2)
%     Tr_calcul_0_1=(1.26 -0.51*zeta+2.58.*zeta^2)/omega_n;
% else
%     Tr_calcul_0_1=(4.67*zeta -1.2)/omega_n;
% end
%

% Peak time,Tp Calculations
[y1,t1]=step(sys1);
Mp=max(y(:,:))-1;
nn=length(y(:,1));
% w=find(nn(1,:)==Mp);%
T=nn(w,:);%
% actual_Tp=answer5(a5,2))
calculated_Mp=100*exp(-pi*zeta/sqrt(1-(zeta*zeta)));

% The decay ratio DR Calculations
Calculated_DR=exp(-2*zeta*pi/sqrt(1-zeta^2));
% The damping factor D_zeta, Calculations
Calculated_D_zeta=exp(zeta*pi/(1-zeta^2));
% The period, TO and frequency of any oscillations in the response, Calculations:
% The period of oscillation
Calculated_To=2*pi/(omega_n*sqrt(1-zeta^2));
% The frequency of oscillation
Calculated_omega= omega_n*sqrt(1-zeta^2);
% % Delay time,Td Calculations
% format bank
% [y1,t1]=step(sys1,0:0.01:7);%
% nn=length(y(:,1));
% for ii=1: nn
%     if y(ii,1)==0.5
%         qq=ii;
%         break
%     end
% end
% actual_Td=nn(qq,2);%
calculated_Td=(1.2378 -0.156*zeta+0.592*zeta^2)/omega_n;
% Time constant,T Calculations
Time_constant=1/(zeta*omega_n);
% Steady state error, Ess Calculations
Ess= 1/(1+ dcgain(sys1));
%DC gain, Calculations
t0=0.001:100; % time and increment
y=step(sys1,t0);
DC_gain2=y(length(t0));
home, printsys(num,den,'s')
disp(' '); sys1; step(sys1)
disp(' ')
disp(' Time constant,T Calculations :')

disp('======================================'

Stability analysis:
' fprintf( ' The system poles, P1 = %2.2f , P2 = %2.2f 
'system_poles(1,1),system_poles(2,1))
' fprintf( ' %s 
',uu), disp(' ')
' fprintf( ' The damping ratio,Zeta = %2.5f  Seconds 
',zeta)
' fprintf( ' The UNdamped natural frequency, Omega_n=  
',omega_n)
' fprintf( ' The Damped natural frequency, Omega_d=  
',omega_d)

% Delay time,Td Calculations
% format bank
% [y1,t1]=step(sys1,0:0.01:7);%
% nn=length(y(:,1));
% for ii=1: nn
%     if y(ii,1)==0.5
%         qq=ii;
%         break
%     end
% end
% actual_Td=nn(qq,2);%
calculated_Td=(1.2378 -0.156*zeta+0.592*zeta^2)/omega_n;
% Time constant,T Calculations
Time_constant=1/(zeta*omega_n);
% Steady state error, Ess Calculations
Ess= 1/(1+ dcgain(sys1));
%DC gain, Calculations
t0=0.001:100; % time and increment
y=step(sys1,t0);
DC_gain2=y(length(t0));
home, printsys(num,den,'s')
disp(' '); sys1; step(sys1)
disp(' ')
disp(' Time constant,T Calculations :')

disp('======================================'

Stability analysis:
' fprintf( ' The system poles, P1 = %2.2f , P2 = %2.2f 
'system_poles(1,1),system_poles(2,1))
' fprintf( ' %s 
',uu), disp(' ')
' fprintf( ' The damping ratio,Zeta = %2.5f  Seconds 
',zeta)
' fprintf( ' The UNdamped natural frequency, Omega_n=  
',omega_n)
' fprintf( ' The Damped natural frequency, Omega_d=  
',omega_d)
Farhan A. Salem

International Journal of Current Engineering and Technology, Vol.3, No.2 (June 2013)

A derivation of accurate analytical expressions for mechatronics systems performance measures with minimum possible deviation at actual values, that can be used to calculate, evaluate and verify the actual mechatronics systems' performance are proposed, also to verify the accuracy and correctness of derived analytical expressions, as well as to return the precise numerical values of performance specifications, a new MATLAB built-in function named ‘mecspec’ is designed and proposed. The proposed expressions were tested for different systems, the resulted numerical values were compared with actual numerical values obtained using MATLAB, the result show that both values estimated and obtained using MATLAB are identical with maximum percentage relative error between 0.002-0.012 The derived analytical expressions and new built-in function are applied for first and second-order systems.
with no zeros and represent the essential qualities of higher-order systems with one or two dominant poles and are intended to be used in systems dynamics analysis, design, control and related sciences, as well as for the application in educational process.

Acknowledgement

Author would like to express his sincere thanks to Anas Ismaiel AbuMahfouz, for his valuable suggestions and help during the development of the current work.

References


http://courses.engr.illinois.edu/ece486/lab/estimates/estimates.html

Author’s Profile:

Farhan Atallah Salem Abu Mahfouz: B.Sc., M.Sc and Ph.D., in Mechatronics of production systems, Moscow state Academy. Now he is ass. Professor in Taif University, Mechatronics program, Dept. of Mechanical Engineering and gen-director of alpha center for engineering studies and technology researches. Research Interests; Design, modeling and analysis of primary Mechatronics Machines. Control and algorithm selection, design and analysis for Mechatronics systems. Rotor Dynamics and Design for Mechatronics applications