Time Delay Handling in Dominant Pole Placement with PID Controllers to Obtain Stability Regions using Random Sampling

Kaushik Halder[1], Saptarshi Das[2, 3], and Amitava Gupta[4]

1) Department of Mechanical Engineering Sciences, University of Surrey, Guildford GU2 7XH, UK.
2) Department of Mathematics, College of Engineering, Mathematics and Physical Sciences, University of Exeter, Penryn Campus, Cornwall TR10 9FE, UK.
3) Institute for Data Science and Artificial Intelligence, University of Exeter, Laver Building, North Park Road, Exeter, Devon EX4 4QE, UK.
4) Department of Power Engineering, Jadavpur University, Salt Lake Campus, LB-8, Sector 3, Kolkata-700098, India.

Author's Emails:
k.halder@surrey.ac.uk (K. Halder)
saptarshi.das@ieee.org, s.das3@exeter.ac.uk (S. Das*)
amitava.gupta@jadavpuruniversity.in (A. Gupta)
Phone number: +44-7448572598

Abstract:

This paper proposes a new formulation of proportional-integral-derivative (PID) controller design using the dominant pole placement method for handling second order plus time delay (SOPTD) systems. The proposed method does not contain any finite term approximation like different orders of Pade for handling the time-delay term, in the quasi-polynomial characteristic equation. Rather it transforms the transcendental exponential delay term of the plant into finite number of discrete-time poles by a suitable choice of the sampling time. The PID controller has been represented by Tustin's discretization method and the PID controller gains are obtained using the dominant pole placement criterion where the plant is discretized using the pole-zero matching method. A random search and optimization method has been used to obtain the stability region in the desired closed loop parameters space by minimising the integral squared error (ISE) criterion by randomly sampling from the stabilizable region and then these closed loop parameters are mapped on to the PID controller gains. Effectiveness of the proposed methodology is shown for nine test-bench plants with different lag to delay ratios and open loop damping levels, and the effect of choosing different sampling times, using credible numerical simulations.

Keywords:

Dominant pole placement; PID controller; SOPTD plant; pole-zero matching; Tustin discretization

1. Introduction

In the last few decades several methods have been proposed to design PID controllers to handle the generic second order plus time delay (SOPTD) processes that can capture both sluggish and oscillatory open loop dynamics (O'Dwyer 2009). Amongst various PID controller design methods, the dominant pole placement is quite challenging for controlling plants with time delays because of the presence of exponential delay term (e^{-Ls}), thus making the system to have an infinite order. This gives rise to quasi-polynomial type characteristic equations which is harder to tackle compared to the rational polynomials as reported in (Silva et al. 2007). The stability region of time delay systems may decrease with the increasing order of the Pade approximation (Bhattacharyya et al. 2009).

1.1 Previous works on dominant pole placement based PID controller design

Previously, few researchers have attempted to design dominant pole placement based PID controllers for handling time-delay systems in continuous time (either using root locus or Nyquist plot) (Wang et al. 2009) and discrete time domains (Dincel & Söylemez 2014). A semi-automated pole placement based state feedback controller has been proposed to shift the real part of the right most root of the

closed loop system to the far possible left-half of the *s*-plane in quasi-continuous way in (Michiels et al. 2002) and (Michiels & Vyhlidal 2005) for neutral and retarded type time delay system respectively. This methodology does not allow direct pole placement for SOPTD process models since these methods only monitor the real parts of the roots. A mixed PI and state feedback controller has been designed for retarded time-delay systems to overcome the above mentioned problem in (Michiels et al. 2010). This method combines both direct pole placement and the spectral abscissa minimization to determine the controller parameters. A dominant pole placement based PID controller has been designed to place three dominant poles of the delayed PID control loop in (Zitek et al. 2019), (Zitek & Fiser 2018) and (Zitek et al. 2013). These methods use dimensional analysis and integral absolute error (IAE) criterion for obtaining a generalized model of delayed PID control loop and satisfactory control performances.

1.2 Previous works on stability regions for handling delayed systems with PID controllers

For analyzing the stability of the time delay systems, several researchers have attempted to find the stability region mainly in the PI/PID controller parameter space. For example, (Silva et al. 2002) have derived admissible range of proportional controller gains using Hermite–Biehler theorem in quasi-polynomial for the first order plus time delay (FOPTD) systems and then each proportional gain within the admissible range is used to obtain the stabilizing set of integral and derivative controller gains in two dimensional space. By deriving the necessary conditions with frequency equals to zero or infinity, the extended Hermite–Biehler theorem have been used to find the admissible ranges of PID controller gains in (Ou et al. 2006). Here, the complete set of stabilising PID controller gains were obtained as special parametric structure i.e. polyhedron by sweeping over the entire range of admissible proportional gain and then found a set of PID controller gains to be at the centre of the largest stability ball inscribed inside the polyhedron for second-order integrating processes with time delay. (Tan 2005) has used the stability boundary locus to obtain the stability region of PI and PID controller parameter space for time delay systems by converting the time delay term in real and imaginary part. A graphical technique based on D-partitioning technique has been used to find out the stability region in PID controller space for handling the time delay systems in (Shafiei & Shenton 1994). Using this graphical stability region criterion in quasi-polynomials, the stability region was obtained in integral-derivative plane for fixed proportional gain value for the time delay systems in (Wang 2012). A geometric approach has been used to find out the stability region in PI/PID controller parameter space for ensuring closed loop stability of time delay systems and fragility of PI and PID controllers in (Morarescu et al. 2011; Méndez-Barrios et al. 2008) respectively. The controller gains are selected inside the stability region which is far away from the boundary of the stability region.

Recently, proportional-delayed controllers were designed for handling linear time invariant systems in (Hernández-Diez et al. 2018) which considered a geometric approach that allows partitioning the parameter space in to regions, defined by separating hyperplane, with constant number of unstable roots. A graphical method was proposed for tuning the PI–PD controller parameters for time delayed unstable systems (Onat 2019). This method is based on the stability boundary locus by equating the real and the imaginary parts of the characteristic equation to zero and uses a concept named convex stability region. The drawbacks of this method is that in the region between the boundaries of the stability locus and the convex region, we cannot find any stable solution using convex optimization. Another study on PI-PD controller tuning for time delay system is proposed in (Hernández-Diez et al. 2019) which uses the graphical method to find out the stability region in the controller parameter space. For the digital implementation of the proposed control scheme in (Hernández-Diez et al. 2019), the PD controller uses the Euler approach for approximating the derivative term which is termed as the Pδ controller. However, all these methods, discussed above are designed to find a stabilizing set of PID controller parameters in 2D space, keeping one controller parameter fixed, especially the proportional gain. (Mesbahi & Haeri 2014) have used sampling-based stability region derivations but used the grid search whereas (Das et al. 2018) used rejection sampling for obtaining the stability region in three-dimensional space which is computationally expensive.

Contrary to the above approaches, in this paper, we have taken a random search and optimization route as an intelligent sampler to uncover the stability region in the joint space of dominant pole placement parameters and controller parameters in order to design dominant pole placement based PID controllers for handing time delay systems. Therefore, our approach is based on random sampling of the feasible stabilizing space while minimizing an objective function. Here, the design parameter space can be multidimensional compared to the previous works where by one controller gain was fixed to find out the others. Due to the nature of the random sampling based discovery of the stability region, our method is capable of approximating the complex shape of the stabilizable set in the controller parameters space, whereas most literature have drawn hard boundaries based on some approximations of the delays using Pade method.

1.3 Novelty of this work

This paper proposes a novel dominant pole placement based PID controller design method to control SOPTD plants which is an extension of the previous works reported in (Saha et al. 2012)(Das et al. 2012)(Das et al. 2018). To eliminate the problem of increased order due to finite term approximation of the delay term, (Wang et al. 2007) have proposed a method of mapping the time-delay as finite number of poles in discrete time domain as $e^{-Ls} = z^{-n}$. This is based on the consideration that the delay (L) is an integer multiple of the sampling time (T_s) i.e. $n = L/T_s, n \in \mathbb{Z}_+$ (Wang et al. 2007)(Ergenc 2012). Unlike the approximations in (Wang et al. 2007) e.g. $L \approx T_s$ to overcome the infinite dimensionality problem of the closed loop time delay systems for the design of dominant pole placement controllers, in practice the choice of T_s should be sufficiently small compared to the open loop time constants (τ) and delays i.e. $T_s << \{\tau, L\}$ which has been considered in this paper.

In this paper, we assume the delay to sampling time ratio being an integer i.e. $L/T_s = n, n \in \mathbb{Z}_+$ which allows handling of finite number of poles in the complex *z*-plane, for designing the PID controller using the pole placement method, instead of using fixed order Pade approximations (Silva et al. 2007). In contrast to the popular third order Pade approximation method to transform the quasi-polynomials into rational ones (Das et al. 2018), here the idea is to use the discretization method to transform the delay term as finite number of poles. Therefore, our proposed method can be extended for systems with large time delays unlike previous approaches. However, the idea of using all real and all complex non-dominant poles in (Das et al. 2018) have also been used here for the plants being discretized using pole-zero matching method while the PID controller has been discretized using the Tustin's method or bilinear transform as recommended in (Franklin et al. 1998). This is because it is not possible for the continuous time PID controller to be discretized using the pole-zero matching method, since at $s = 0$, the DC gain of the PID controller becomes infinite. To overcome this problem, we use the Tustin's discretization method to transform the PID controller poles/zeros in the discrete-time domain from its continuous-time version. Next, in order to determine discrete time PID controller gains for SOPTD plants, the coefficient matching method (Kiong et al. 1999; Åström & Hägglund 2006) has been used. The obtained discrete time PID controller gains are then mapped back to its continuous time version (Wang et al. 2007), which can be used to control the continuous-time SOPTD plant. In this way, we get a scenario where as if the whole design method was carried out in continuous time domain. Here, the discretization for both the controller and the plant is used to transform the infinite-order quasi-polynomial in *s*-domain so that it can be easily handled as rational polynomials in the *z*-domain.

2. Theoretical formulation

This section presents the design methodology of dominant pole placement based PID controller to handle SOPTD systems using the coefficient matching method which has been adopted from (Kiong et al. 1999; Åström & Hägglund 2006).

2.1 Discretization of SOPTD system using pole-zero matching method

In order to design the dominant pole placement based PID controller, the continuous time SOPTD system is discretized using the pole-zero matching method (Franklin et al. 1998) which can be described as the following steps:

Step 1: Transformation of zeros and poles of SOPTD system from *s*-domain to its equivalent *z*-domain using $z = e^{sT_s}$ and delay term $e^{-Ls} = z^{-n}, n = L/T_s \in \mathbb{Z}_+$.

Step 2: Poles/zeros at $s = -\infty$ are mapped at $z = 0$ and poles/zeros at $s = 0$ are mapped at $z = 1$.

Step 3: DC Gain (*K*) of the open loop system in $s \leftrightarrow z$ domain has to be the matched as:

$$G(s)\big|_{s=0} = G(z)\big|_{z=1}. \tag{1}$$

Let us consider the continuous time open-loop SOPTD system as:

$$G(s) = Ke^{-Ls}/(\tau^2 s^2 + 2\zeta_{ol}\tau s + 1), \tag{2}$$

where, $\{K, L, \tau, \zeta_{ol}\}$ define the DC gain, time delay, time constant and damping ratio respectively.

The system (2) can be controlled by the continuous time PID controller which can be represented as:

$$C(s) = K_p + K_i/s + K_d s, \tag{3}$$

where, $\{K_p, K_i, K_d\}$ represents the proportional, integral and derivative gains.

It is seen that excluding the delay part in (2), the plant has two open-loop poles, which are located at:

$$s_{1,2} = \left(-\zeta_{ol} \pm j\sqrt{1-\zeta_{ol}^2}\right)/\tau. \tag{4}$$

Expressing the open loop natural frequency (ω_{ol}) in the form of time constant $\left(\omega_{ol} = 1/\tau\right)$, equation (4) can be rewritten as:

$$s_{1,2} = -\zeta_{ol}\omega_{ol} \pm j\omega_{ol}\sqrt{1-\zeta_{ol}^2}. \tag{5}$$

Now discretizing (2) to map the open loop continuous time poles in discrete time domain with the specified sampling time (T_s) as in (Franklin et al. 1998), yields:

$$z_1 = e^{s_1 T_s}, z_2 = e^{s_2 T_s}. \tag{6}$$

Using (6) and approximating the time delay term as in *step 1* i.e. $e^{-Ls} = z^{-n}$, the corresponding discretized open loop transfer function of (2) can be represented as:

$$G(z) = \tilde{K}/\left(z^n (z - z_1)(z - z_2)\right), \tag{7}$$

where, $\tilde{K}$ is the equivalent static gain of the corresponding open-loop discrete time system. The value of $\tilde{K}$, described in *step 3*, can be obtained using the pole-zero matching method (Franklin et al. 1998) as:

$$Ke^{-Ls}/(\tau^2 s^2 + 2\zeta_{ol}\tau s + 1)\big|_{s=0} = \tilde{K}/\left(z^n (z - z_1)(z - z_2)\right)\big|_{z=1}. \tag{8}$$

Therefore, we get:

$$\tilde{K} = K(1 - z_1)(1 - z_2). \tag{9}$$

2.2 Discrete time PID controller design using Tustin's method

Here, the continuous time PID controller (3) has been discretized using Tustin's method or bilinear transformation with specified T_s:

$$s \to \left((2/T_s)\left((1-z^{-1}) / (1+z^{-1}) \right) \right). \tag{10}$$

Substituting (10) in (3), the equivalent discrete time PID controller can be written as:

$$C(z) = \frac{\left((4K_d + 2K_pT_s + K_iT_s^2)z^2 + (2K_iT_s^2 - 8K_d)z + (4K_d - 2K_pT_s + K_iT_s^2) \right)}{2T_s(z^2-1)}. \tag{11}$$

Now, closed-loop characteristic polynomial for the discretized system (7) with the discrete PID controller (11) can be represented as:

$$1 + G_{ol}(z) = 1 + C(z)G(z) = 0, \tag{12}$$

which implies,

$$\left(2T_s z^n (z-z_1)(z-z_2)(z^2-1) + \tilde{K}\left\{ (4K_d + 2K_pT_s + K_iT_s^2)z^2 + (2K_iT_s^2 - 8K_d)z + (4K_d - 2K_pT_s + K_iT_s^2) \right\} \right) = 0. \tag{13}$$

It can be seen that the order of the polynomial in (13) is (n+4) which means equation (13) has (n+4) number of roots or closed loop poles. Also, it can be determined that the closed-loop zeros and poles can be modulated by the selection of PID controller gains $\{K_p, K_i, K_d\}$ and the sampling time (T_s). For obtaining the PID controller gains using co-efficient matching method, here we have chosen (n+4)th order desired closed loop characteristic polynomial where the two specified poles are dominant and the rest are non-dominant in nature.

2.3 Deriving expressions for stabilizing PID controllers

Next, the analytical forms of the discrete time PID controller gains $\{K_p, K_i, K_d\}$ are obtained by matching the coefficients of the discrete-time finite-term characteristic polynomial for the given set of open loop process parameters $\{K, L, \tau, \zeta_{ol}, \omega_{ol}\}$ vs. the desired characteristic polynomial which contains the user-defined closed-loop specifications $\{m, \zeta_{cl}, \omega_{cl}\}$ as discussed in (Kiong et al. 1999; Åström & Hägglund 2006). Here, for the $(n+4)^{th}$ order closed loop process in (13), the PID controller gains $\{K_p, K_i, K_d\}$ are found to be unique because these gains $\{K_p, K_i, K_d\}$ are associated only with the order 0 to 2 of the discrete-time operator 'z'. So, only the coefficients of (z^0, z^1, z^2) in (13) are required to compare with the same coefficients of the desired closed-loop characteristic polynomial, in order to determine the three PID controller gains $\{K_p, K_i, K_d\}$, using the coefficient matching method. Since the closed loop polynomial is of $(n+4)^{th}$ order, the desired closed loop characteristic equation should also contain two dominant and $(n+2)$ number of non-dominant poles. However, the desired characteristic polynomial can be designed in two different ways depending on the nature of the non-dominant poles e.g. all real, all complex conjugate following the method in (Das et al. 2018). Again, in the case of all complex conjugate non-dominant poles, there may be two additional cases depending on the influence of the pole placement parameter (m) on either both real and imaginary parts or only on the real part. The detailed derivations for these three cases to obtain the PID controller gains are shown in the following section.

3. Dominant pole-placement based PID controller design methodology

3.1 All real non-dominant poles

In the first case, to frame a dominant pole placement with the discrete time PID controller gains $\{K_p, K_i, K_d\}$, the dominant complex conjugate $(s_{1,2}^d)$ and the non-dominant real (s_3^{nd}) poles of the desired closed-loop characteristic polynomial in continuous time can be considered as:

$$s_{1,2}^{d}=\left(-\zeta_{cl}\omega_{cl}\pm j\omega_{cl}\sqrt{1-\zeta_{cl}^{2}}\right),s_{3}^{nd}=-m\zeta_{cl}\omega_{cl}. \tag{14}$$

Now, discretizing (14) with the sampling time (T_s) yields a mapping of the poles from $s\to z$ plane as:

$$z_{1}^{d}=e^{s_{1}^{d}T_{s}},z_{2}^{d}=e^{s_{2}^{d}T_{s}},z_{3}^{nd}=e^{s_{3}^{nd}T_{s}}. \tag{15}$$

Then using (15), the $(n+4)^{th}$ order discretized desired characteristic polynomial with two dominant poles and the rest being non-dominant real poles can be expressed as:

$$\Delta(z)=\left(z-z_{1}^{d}\right)\left(z-z_{2}^{d}\right)\left(z-z_{3}^{nd}\right)^{n+2}=0. \tag{16}$$

It is evident that (16) has no dependency on *n* being even or odd, unlike the latter cases. Again, using binomial expansion containing non-dominant poles, equation (16) yields:

$$\begin{aligned}
&z^{n+4}\left[\binom{n+2}{0}\left(-z_{3}^{nd}\right)^{0}\right]+\\
&z^{n+3}\left[\binom{n+2}{1}\left(-z_{3}^{nd}\right)^{1}-\binom{n+2}{0}\left(z_{1}^{d}+z_{2}^{d}\right)\left(-z_{3}^{nd}\right)^{0}\right]+\cdots+\\
&z^{2}\left[\binom{n+2}{n+2}\left(-z_{3}^{nd}\right)^{n+2}-\binom{n+2}{n+1}\left(-z_{3}^{nd}\right)^{n+1}\left(z_{1}^{d}+z_{2}^{d}\right)+\binom{n+2}{n}\left(-z_{3}^{nd}\right)^{n}z_{1}^{d}z_{2}^{d}\right]+\\
&z^{1}\left[-\binom{n+2}{n+2}\left(-z_{3}^{nd}\right)^{n+2}\left(z_{1}^{d}+z_{2}^{d}\right)+\binom{n+2}{n+1}\left(-z_{3}^{nd}\right)^{n+1}z_{1}^{d}z_{2}^{d}\right]+\\
&z^{0}\left[\binom{n+2}{n+2}\left(-z_{3}^{nd}\right)^{n+2}z_{1}^{d}z_{2}^{d}\right]=0,
\end{aligned} \tag{17}$$

where, $\binom{n}{r}={}^{n}C_{r}=n!/\left(r!(n-r)!\right)$ is the binomial coefficient.

Now, using the coefficients $A_j, j=\{(n+4),(n+3),\cdots,2,1,0\}$, which can easily be obtained from the open-loop system parameters $\{K,L,\tau,\zeta_{ol},\omega_{ol}\}$ and the desired closed-loop parameters $\{m,\zeta_{cl},\omega_{cl}\}$ equation (17) can be represented as:

$$\Delta_{1}(z)=A_{n+4}z^{n+4}+A_{n+3}z^{n+3}+\cdots+A_{2}z^{2}+A_{1}z^{1}+A_{0}z^{0}=0. \tag{18}$$

Now, matching the coefficients of z^0, z^1, z^2 in both the equations (13) and (18) yields:

$$\begin{aligned}
&z^{0}:\tilde{K}\left(4K_{d}-2K_{p}T_{s}+K_{i}T_{s}^{2}\right)=A_{0}\\
&z^{1}:\tilde{K}\left(2K_{i}T_{s}^{2}-8K_{d}\right)=A_{1}\\
&z^{2}:\tilde{K}\left(4K_{d}+2K_{p}T_{s}+K_{i}T_{s}^{2}\right)=A_{2}.
\end{aligned} \tag{19}$$

Solving (19) yields the three PID controller gains in explicit cascaded form as:

$$K_{i}=\frac{(A_{0}+A_{1}+A_{2})}{4\tilde{K}T_{s}^{2}},K_{d}=\frac{K_{i}T_{s}^{2}}{4}-\frac{A_{1}}{8\tilde{K}},K_{p}=\frac{1}{2T_{s}}\left(4K_{d}+K_{i}T_{s}^{2}-\frac{A_{0}}{\tilde{K}}\right). \tag{20}$$

It is seen from the actual closed-loop characteristics polynomial (13) that the three PID controller gains (K_p, K_i, K_d) are connected only with the three coefficients z^0 to z^2 and rest of the coefficients i.e. z^3 to z^{n+4} do not contain any PID controller parameters. Thus, it is logical to compare only these three coefficients (z^0 to z^2) of the actual closed loop polynomial (13) with the desired closed loop polynomial (18) to obtain the analytical expression of PID controller parameters. However, it is also seen that the

three controller expressions (20) are the function of all chosen dominant and non-dominant poles. Therefore, it is true that the designed three controller parameters (K_p, K_i, K_d) will guarantee the position of the dominant poles and the non-dominant ones are safely placed far away from the imaginary axis so that they do not affect the dominant closed loop dynamics.

3.2 All complex conjugate non-dominant poles when m is in both the real and imaginary parts

In this subsection, to ensure dominant pole placement, all the non-dominant poles are considered as complex conjugates. However, this assumption is possible only for the even order discretized closed-loop characteristic polynomial (13) i.e. $n/2 \in \mathbb{Z}_+$. In this case, the non-dominant complex conjugate poles may be described as two types i.e. in the first case the pole placement parameter (m) being connected with both the real and imaginary parts. In the next case, the pole placement parameter (m) only affects the real part of the complex conjugate non-dominant closed-loop poles in the continuous time domain. In the first case, the non-dominant complex conjugate poles take the form:

$$s_{4,5}^{nd} = m\left(-\zeta_{cl}\omega_{cl} \pm j\omega_{cl}\sqrt{1-\zeta_{cl}^2}\right). \tag{21}$$

Then its discrete equivalent using (T_s) can be represented as:

$$z_4^{nd} = e^{s_4^{nd}T_s}, z_5^{nd} = e^{s_5^{nd}T_s}. \tag{22}$$

Now, using (15) and (22), the discretized desired closed loop characteristic polynomial becomes:

$$\Delta(z) = \left(z - z_1^d\right)\left(z - z_2^d\right)\left(z - z_4^{nd}\right)^{\frac{n+2}{2}}\left(z - z_5^{nd}\right)^{\frac{n+2}{2}} = 0. \tag{23}$$

It is to be noted that to obtain the $(n+4)^{\text{th}}$ order desired characteristic polynomial (23) with two dominant poles (15) and rest being non-dominant complex conjugate poles (22), the non-dominant terms $\left(z - z^{nd}\right)^{n+2}$ in (23) is partitioned with $\left(z - z_4^{nd}\right)^{\frac{n+2}{2}}\left(z - z_5^{nd}\right)^{\frac{n+2}{2}}$ unlike the case of all real non-dominant pole type in (16). Under such scenario, the order of the desired characteristics polynomial (23) i.e. order of the closed loop discrete time system must be even by satisfying $n/2 \in \mathbb{Z}_+$.

Now, using binomial expansion containing the non-dominant poles, equation (23) yields:

$$z^{n+4}\left[\binom{\frac{n+2}{2}}{0}^2\left(-z_4^{nd}\right)^0\left(-z_5^{nd}\right)^0\right] +$$

$$z^{n+3}\left[-\binom{\frac{n+2}{2}}{0}^2\left(z_1^d + z_2^d\right) + \binom{\frac{n+2}{2}}{0}\binom{\frac{n+2}{2}}{1}\left(\left(-z_4^{nd}\right)^0\left(-z_5^{nd}\right)^1 + \left(-z_4^{nd}\right)^1\left(-z_5^{nd}\right)^0\right)\right]$$

$$+\cdots+$$

$$z^2\left[\binom{\frac{n+2}{2}}{\frac{n-2}{2}}\binom{\frac{n+2}{2}}{\frac{n+2}{2}} z_1^d z_2^d \times\left(\left(-z_4^{nd}\right)^{\frac{n-2}{2}}\left(-z_5^{nd}\right)^{\frac{n+2}{2}} + \left(-z_4^{nd}\right)^{\frac{n+2}{2}}\left(-z_5^{nd}\right)^{\frac{n-2}{2}}\right)\right.$$

$$+\binom{\frac{n+2}{2}}{\frac{n}{2}}^2 z_1^d z_2^d\left(-z_4^{nd}\right)^{\frac{n}{2}}\left(-z_5^{nd}\right)^{\frac{n}{2}} - \binom{\frac{n+2}{2}}{\frac{n}{2}}\binom{\frac{n+2}{2}}{\frac{n+2}{2}}\left(z_1^d + z_2^d\right) +$$

$$\left.\times\left(\left(-z_4^{nd}\right)^{\frac{n}{2}}\left(-z_5^{nd}\right)^{\frac{n+2}{2}} + \left(-z_4^{nd}\right)^{\frac{n+2}{2}}\left(-z_5^{nd}\right)^{\frac{n}{2}}\right) + \binom{\frac{n+2}{2}}{\frac{n+2}{2}}^2\left(-z_4^{nd}\right)^{\frac{n+2}{2}}\left(-z_5^{nd}\right)^{\frac{n+2}{2}}\right]$$

$$z^1\left[\begin{array}{l}\left(\begin{array}{c}\frac{n+2}{2}\\ \frac{n}{2}\end{array}\right)\left(\begin{array}{c}\frac{n+2}{2}\\ \frac{n+2}{2}\end{array}\right)z_1^d z_2^d\times\left(\left(-z_4^{nd}\right)^{\frac{n}{2}}\left(-z_5^{nd}\right)^{\frac{n+2}{2}}+\left(-z_4^{nd}\right)^{\frac{n+2}{2}}\left(-z_5^{nd}\right)^{\frac{n}{2}}\right)\\ -\left(\begin{array}{c}\frac{n+2}{2}\\ \frac{n+2}{2}\end{array}\right)^2\left(-z_4^{nd}\right)^{\frac{n+2}{2}}\left(-z_5^{nd}\right)^{\frac{n+2}{2}}\left(z_1^d+z_2^d\right)\end{array}\right]+$$

$$z^0\left[\left(\begin{array}{c}\frac{n+2}{2}\\ \frac{n+2}{2}\end{array}\right)^2 z_1^d z_2^d\left(-z_4^{nd}\right)^{\frac{n+2}{2}}\left(-z_5^{nd}\right)^{\frac{n+2}{2}}\right]=0\,. \tag{24}$$

Similar to the previous case, using the coefficients $A'_j, j=\{(n+4),(n+3),\cdots,2,1,0\}$ (where A'_j can be obtained from open loop $\{K,L,\tau,\zeta_{ol},\omega_{ol}\}$ and desired closed loop $\{m,\zeta_{cl},\omega_{cl}\}$ parameters), equation (24) can be written as:

$$\Delta_2(z)=A'_{n+4}z^{n+4}+A'_{n+3}z^{n+3}+\cdots+A'_2z^2+A'_1z^1+A'_0z^0=0\,. \tag{25}$$

Now matching the coefficients of (13) and (25), the following set of equations are obtained:

$$\begin{aligned}&z^0:\tilde{K}\left(4K_d-2K_pT_s+K_i\right)=A'_0\\ &z^1:\tilde{K}\left(2K_iT_s^2-8K_d\right)=A'_1\\ &z^2:\tilde{K}\left(4K_d+2K_pT_s+K_iT_s^2\right)=A'_2.\end{aligned} \tag{26}$$

Solving (26) for three different powers of the discrete time operator z i.e. z^0 to z^2 yields the PID controller gains in cascaded explicit form as:

$$K_i=\frac{\left(A'_0+A'_1+A'_2\right)}{4\tilde{K}T_s^2},K_d=\frac{K_iT_s^2}{4}-\frac{A'_1}{8\tilde{K}},K_p=\frac{1}{2T_s}\left(4K_d+K_iT_s^2-\frac{A'_0}{\tilde{K}}\right). \tag{27}$$

3.3 All complex conjugate non-dominant poles when m is only in the real part

Similar to the above subsection, here the non-dominant complex conjugate closed-loop poles in continuous time can be described as:

$$s_{6,7}^{nd}=-m\zeta_{cl}\omega_{cl}\pm j\omega_{cl}\sqrt{1-\zeta_{cl}^2}\,, \tag{28}$$

where *m* is connected with the real part only. Then its discrete-time equivalent can be obtained as:

$$z_6^{nd}=e^{s_6^{nd}T_s},z_7^{nd}=e^{s_7^{nd}T_s}. \tag{29}$$

Now using (15) and (29), the corresponding desired closed loop characteristic equation becomes:

$$\Delta(z)=\left(z-z_1^d\right)\left(z-z_2^d\right)\left(z-z_6^{nd}\right)^{\frac{n+2}{2}}\left(z-z_7^{nd}\right)^{\frac{n+2}{2}}=0. \tag{30}$$

It is evident that both the expressions in (23) and (30) are exactly the same, except the choice of non-dominant pole locations i.e. instead of (z_4,z_5) the non-dominant poles will be (z_6,z_7), where the pole placement parameter (*m*) is associated with the real part only instead of being in both the real and

imaginary parts. Similar to (25), using the new coefficient $\left(\tilde{A}_j'\right)$ instead of $\left(A_j'\right)$, the equation (30) can be written as:

$$\Delta_2(z)=\tilde{A}_{n+4}'z^{n+4}+\tilde{A}_{n+3}'z^{n+3}+\cdots+\tilde{A}_2'z^2+\tilde{A}_1'z^1+\tilde{A}_0'z^0=0. \tag{31}$$

Now matching the coefficients of (13) and (31), the following set of equations are obtained:

$$\begin{aligned} z^0&:\tilde{K}\left(4K_d-2K_pT_s+K_i\right)=\tilde{A}_0'\\ z^1&:\tilde{K}\left(2K_iT_s^2-8K_d\right)=\tilde{A}_1'\\ z^2&:\tilde{K}\left(4K_d+2K_pT_s+K_iT_s^2\right)=\tilde{A}_2'. \end{aligned} \tag{32}$$

Solving (32) for three different powers of the discrete time operator z i.e. z^0 to z^2 yields the PID controller gains in similar cascaded explicit form like before as:

$$K_i=\frac{(\tilde{A}_0'+\tilde{A}_1'+\tilde{A}_2')}{4\tilde{K}T_s^2},K_d=\frac{K_iT_s^2}{4}-\frac{\tilde{A}_1'}{8\tilde{K}},K_p=\frac{1}{2T_s}\left(4K_d+K_iT_s^2-\frac{\tilde{A}_0'}{\tilde{K}}\right). \tag{33}$$

4. Performance measures of the PID control loops

In industrial process control involving PID control system design problems, the control loop performances are also of major concern apart from the stability (Das et al. 2018). Using the above three different expressions of PID controller gains, here we calculate different performance measures of the control loop. From Figure 1, it is seen for the three different inputs, nine transfer functions play an important role for guaranteeing internal stability and affects different performance measures (Doyle et al. 2013)(Das et al. 2018)(Herreros et al. 2002). Amongst these nine, the following four transfer functions play the key role to characterize the control system performances *viz.* sensitivity $S_e(s)$, complementary sensitivity $T(s)$, control sensitivity $S_u(s)$ and disturbance sensitivity $S_d(s)$ (Das et al. 2018; Das & Pan 2014)(Doyle et al. 2013)(Herreros et al. 2002) as follows:

$$\begin{aligned} &S_e(s)=1/\left(1+G_{ol}(s)\right)=G_{re}=G_{dx_2}=G_{nx_3},\\ &T(s)=G_{ol}(s)/\left(1+G_{ol}(s)\right)=G_{rx_3},S_e(s)+T(s)=1,\\ &S_u(s)=C(s)/\left(1+G_{ol}(s)\right),S_d(s)=G(s)/\left(1+G_{ol}(s)\right), \end{aligned} \tag{34}$$

where, $G_{ol}(s)=C(s)G(s)$ is the open loop transfer function.

It is evident that the standard PID controller without derivative filter makes the control sensitivity transfer function improper with the number of poles less than the number of zeros which prohibit direct calculation using MATLAB's '*step*' command. As reported in (Das et al. 2018), the control signal can be computed with '*impulse*' input to S_u/s when S_u is improper which is an alternative approach and has been used in this paper.

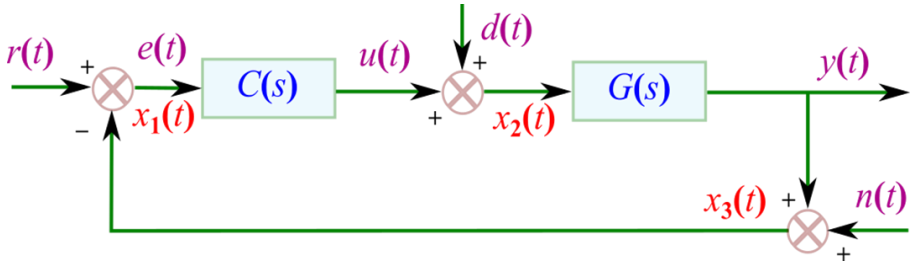

Figure 1: Schematic diagram of SOPTD plant $G(s)$ with PID controller $C(s)$.

5. Simulation and results

5.1 PID controller design using optimization of closed loop design specifications

Here, in order to analyze the effectiveness of the proposed methodology using three different PID controller expressions, we consider nine test-bench plants of different types viz. lag-dominant (G_1-G_3) ($L < \tau$), balanced (G_4-G_6) ($L \approx \tau$) and delay dominant (G_7-G_9) ($L > \tau$) plants which are also further separated by three different damping-ratios for representing different levels of open-loop oscillations i.e. under-damped ($\zeta_{ol} < 1$), critically-damped ($\zeta_{ol} = 1$), and over-damped ($\zeta_{ol} > 1$), respectively. The detailed characteristics and open-loop parameters of each plant are described in (Das et al. 2018) and given in Table 1.

Table 1:Parameters and characteristics of nine test-bench plants under control

Plant	Type	K	L	τ	ζ_{ol}	$\omega_{ol} = 1/\tau$	L/τ
G_1 (Das et al. 2018)	Lag-dominant under damped	1	1	3	0.4	0.3333	0.3333<1
G_2 (Das et al. 2018)	Lag-dominant critically damped	1	0.8	1	1	1	0.8<1
G_3 (Das et al. 2018)	Lag-dominant over damped	1	2	6.324	1.106	0.1581	0.3162<1
G_4 (Das et al. 2018)	Balanced under damped	0.5	1	1	0.6	1	1
G_5 (Das et al. 2018)	Balanced critically damped	1	1	1	1	1	1
G_6 (Das et al. 2018)	Balanced over damped	1	3	3	4	0.3333	1
G_7 (Wang & Shao 2000)	Delay dominant under damped	0.327	1.28	1.025	0.504	0.9749	1.2479>1
G_8 (Das et al. 2018)	Delay dominant critically damped	1	10	1	1	1	10>1
G_9 (Das et al. 2018)	Delay dominant over damped	0.806	2	0.311	1.723	3.2144	6.4288>1

The three different PID controller expressions (20), (27) and (33) have been used to determine the PID controller gains using the specified sampling time T_s for nine test-bench plants. However, based on the choice of T_s, the optimum closed-loop design parameters may vary widely. Again, it may not always be possible to get a stabilizing solution for certain desired specifications in time/frequency domain (Das & Pan 2014)(Pan & Das 2013). Therefore, using a random search and optimization technique, we may determine the three-desired closed-loop performance parameters $\{m, \zeta_{cl}, \omega_{cl}\}$. The particle swarm optimization (PSO) algorithm is used here to determine the closed-loop performance parameters due to its faster capability of exploring a large parameter space for complicated objective function landscape and also filter out the feasible or stability region in the 3D space of three design parameters and the associated PID controller gains which is computationally advantageous over the rejection sampling approach in a blind Monte Carlo simulation framework adopted in (Das et al. 2018). The optimum values of three design parameter $\theta_{opt} = \{m, \zeta_{cl}, \omega_{cl}\}$ are obtained using a PSO based search and optimization algorithm when the integral of squared error (ISE):

$$\theta_{opt} = \arg\min_{\theta \in \Theta} J(\theta),\ J(\theta) = \int_0^T \left(r - y(\theta)\right)^2 dt, \tag{35}$$

criterion is minimized with three different non-dominant pole types for each test-bench plant. The obtained optimum values of user-specified parameters $\{m, \zeta_{cl}, \omega_{cl}\}$ can then be used to find out the PID controller gains $\{K_p, K_i, K_d\}$ which are given in Table 2, along with the plant types and the number of objective function evaluations (N_{eval}). The 3D-stability region in the design and controller parameter space with different non-dominant pole types for each test-bench plants are shown in Figure 2-Figure 3 which were obtained from the PSO algorithm with swarm size of $N_{\text{particle}} = 300$. The parameter search ranges for (35) are considered as $\Theta = \{m \in [0,20], \zeta_{cl} \in [0,5], \omega_{cl} \in [0,20]\}$. Also, a penalty method is adopted to avoid search in the infeasible regions of the objective function where ISE value is too large i.e. ISE $\approx 10^6$, which indicates an unstable or highly oscillatory closed loop response. The stability regions in Figure 2 are obtained via random sampling of the parameter space with PSO algorithm using

the ISE cost function (35) for a step input in the set-point. However, a similar stability region could also have been obtained using the ISE cost function corresponding to disturbance inputs.

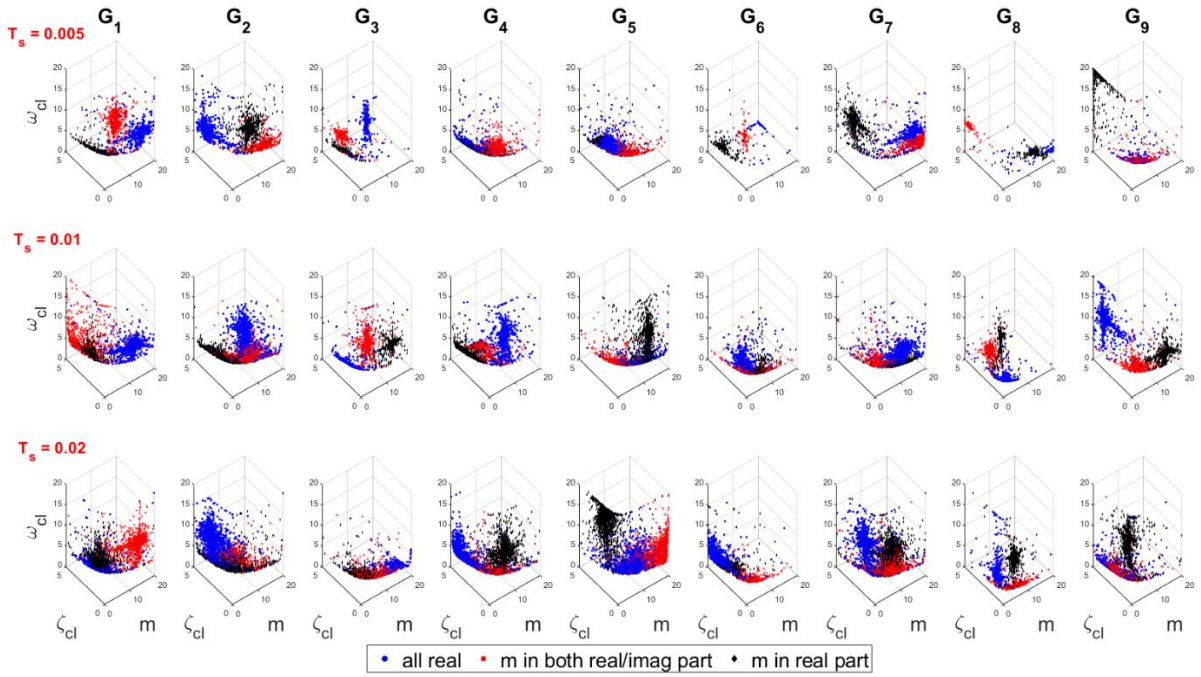

Figure 2: Stability region in the design parameter $\{m, \zeta_{cl}, \omega_{cl}\}$ space for the nine test-bench plants and different sampling times. A threshold (ISE<4800) has been applied on the sampled points. Different colours represent the sampled data-points in the stability regions using the three classes of pole placement algorithms indicated in the legends.

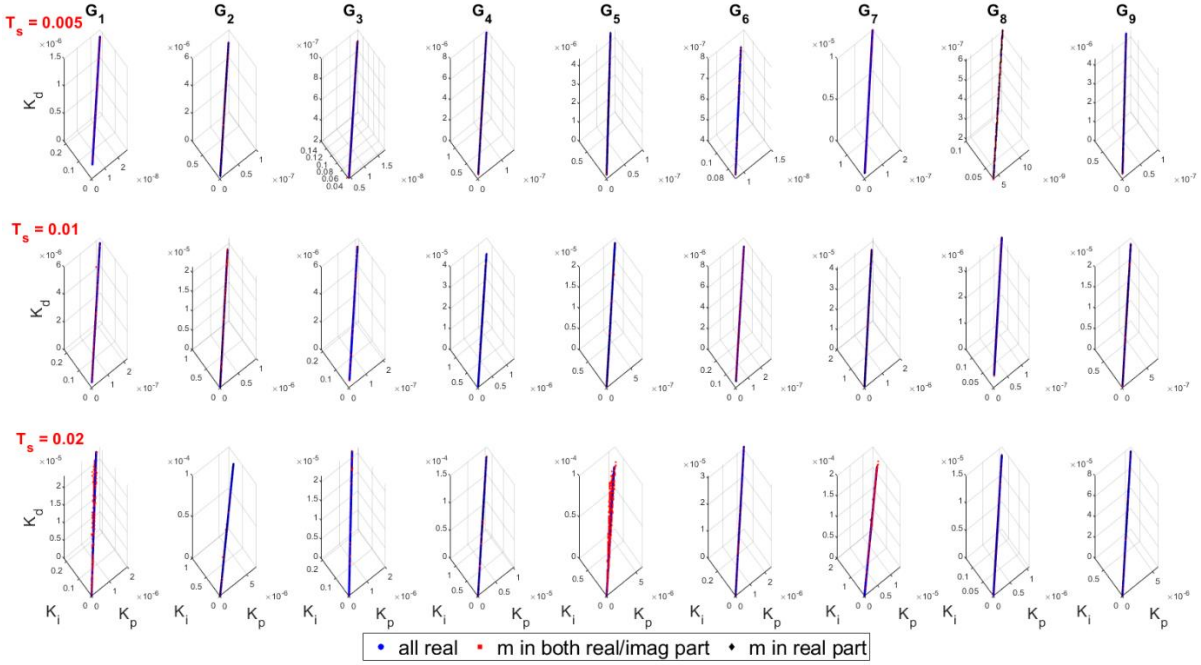

Figure 3: Stability region in the controller parameter space $\{K_p, K_i, K_d\}$ for the nine test-bench plants and different sampling times. A threshold (ISE<4800) has been applied on the sampled points.

The greatest common divisor (GCD) of the delay terms in Table 1 has been used to select a common sampling time so that it is divisible by a common factor for all the 9 test-bench plants as T_s = 0.04 sec, such that the delay to sampling time ratio ($n = L/T_s$) becomes an integer. However, we have derived

the PID controller expressions with non-dominant complex conjugate poles for even-ordered characteristic polynomial i.e. $n/2 \in \mathbb{Z}_+$. Therefore, we have chosen different sampling times T_s = {0.005, 0.01, 0.02}sec by satisfying the condition $T_s \leq 0.04$s and $n/2 \in \mathbb{Z}_+$ for showing its effect on the control loop performances of the SOPTD systems under control. Amongst the three different T_s this paper selects T_s = 0.02 sec for all the nine test-bench plants since it yield the minimum ISE criterion ($J_{\min}$) as shown in the box-plots in Figure 4 across plants and over different sampling times to choose the best T_s for this application. Moreover, during the PSO based random search and optimization for sampling from the feasible stability region, we have simulated the closed loop system using a 9th order Pade approximation for the transcendental equation of the closed loop system. In this paper, the stability derivations do not rely on any approximation of the delay unlike the previous works (Das et al. 2018) where a third order Pade approximation was used in the stability derivations, followed by a random sampling approach. This is required to filter out the unstable closed loop solutions within the random search process within PSO's objective function evaluation, if there is any unstable pole upon Pade approximation of the whole closed loop system for a certain set of controller gains including the higher order effects of the delay term.

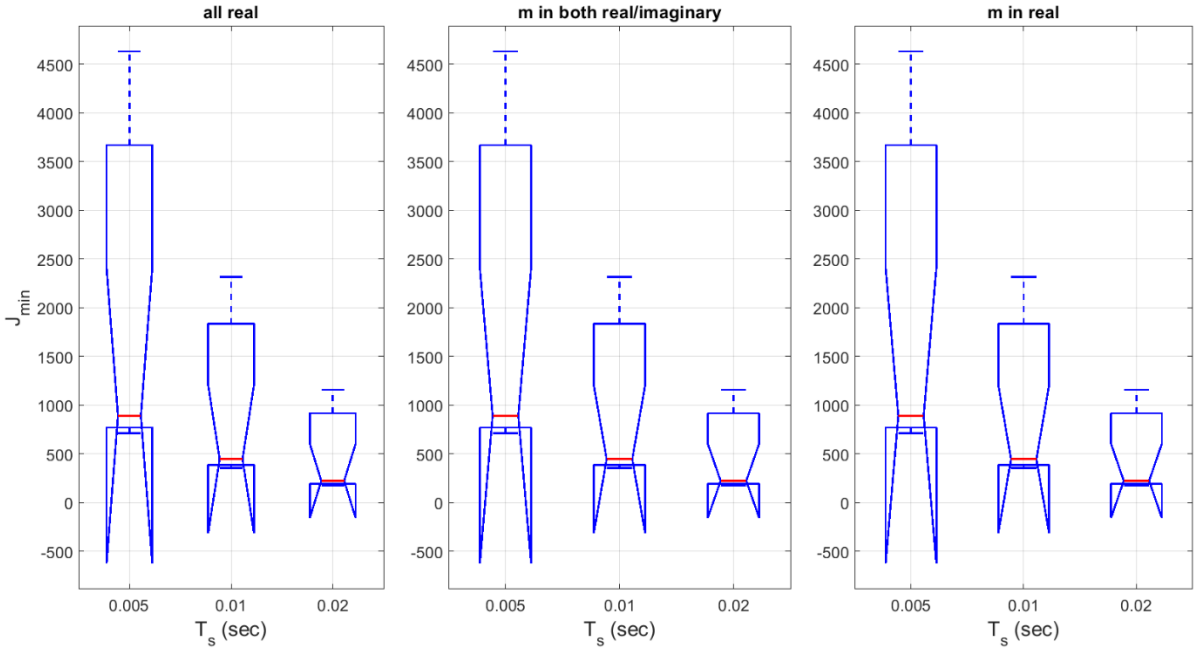

Figure 4: Minimum ISE cost function (J_{min}) with different sampling time (T_s) for nine test bench plants using three different PID controllers obtained from three different non-dominant pole types. Each boxplot shows the statistics across the nine test-bench plants.

It is interesting to note that although the stability regions are found at distinctly different locations in the design parameter space $\{m, \zeta_{cl}, \omega_{cl}\}$ in Figure 2, when they are mapped on to the controller parameter space $\{K_p, K_i, K_d\}$ in Figure 3, they follow almost a straight line for the three cases of sampling times $T_s = \{0.005, 0.01, 0.02\}$ sec which is different from previous works described in section 1.2. These stability regions are intrinsically different from the regular shapes obtained in (Bhattacharyya et al. 2009; Silva et al. 2007) found using analytical methods. However, here we use a random PSO based sampling or search and optimization method to uncover the hidden patterns of the stability region using a threshold (ISE<4800) on all the sampled ISE values, visited by the particles (Pan et al. 2011). The obtained minumum ISE values (J_{min}) in Table 2 from PSO based random search and optimization algorithm for nine test-bench plants with three different PID controller expressions and sampling times are used to choose the best sampling time for rest of the simulation results. Figure 4 shows the box-plot of J_{min} vs. T_s for all nine test-bench plants with the three different PID controllers obtained from the three different non-dominant pole types. It is seen that the values of J_{min} are almost same for three

different non-dominant pole types with same sampling time and their median values are decrecesing with the larger value of T_s compared to smaller one for all three different non-dominant pole types. Therefore, we have selected the sampling time as $T_s = 0.02$ sec yielding the minimum ISE value to show the efficacy of the proposed methodology.

Table 2: PSO based selection of optimum closed loop design parameters and controller parameters at different T_s for nine test-bench plants

Plant	Nondominant pole type	T_s (sec)	ISE	N_{eval}	m	ζ_{cl}	ω_{cl}	K_p	K_i	K_d
		0.005	1487.46	15918	16.0623	0.2568	8.2407	0.00000002	0.12462740	0.00000079
		0.01	743.73	21356	13.2818	0.3006	7.4728	0.00000013	0.12462763	0.00000321
	All real	0.02	371.86	19324	4.3607	2.5405	2.3045	0.00000102	0.12462838	0.00001305
		0.005	1487.46	18658	5.5959	0.3924	15.3917	0.00000002	0.12462934	0.00000079
		0.01	743.73	18334	1.5817	4.8462	3.8605	0.00000013	0.12448930	0.00000320
	m in both real/imaginary	0.02	371.86	21370	14.4470	0.1666	10.6461	0.00000097	0.12462829	0.00001187
		0.005	1487.46	25721	7.1589	2.3498	2.0192	0.00000002	0.12462740	0.00000079
		0.01	743.73	17765	1.9883	2.7253	5.4630	0.00000013	0.12462758	0.00000321
G_1	*m* in real	0.02	371.86	22866	2.7510	1.2696	7.2751	0.00000102	0.12462839	0.00001304
		0.005	743.18	20620	1.4359	3.9174	6.6913	0.00000004	0.34216861	0.00000218
		0.01	371.59	20524	5.6485	0.4011	14.4150	0.00000035	0.34217031	0.00000886
	All real	0.02	185.79	18929	0.6954	3.2629	11.4256	0.00000281	0.34217858	0.00003612
		0.005	743.18	17333	13.9068	0.5150	5.2783	0.00000004	0.34216852	0.00000218
	m in both real/ imaginary	0.01	371.59	20502	8.8372	0.6373	5.7739	0.00000035	0.34217031	0.00000888
		0.02	185.79	15444	11.9460	4.3698	0.5345	0.00000280	0.34217910	0.00003588
		0.005	743.18	8675	7.4931	0.4346	11.6362	0.00000004	0.34216844	0.00000218
		0.01	371.59	25131	5.7417	2.1495	2.6469	0.00000035	0.34217028	0.00000886
G_2	*m* in real	0.02	185.79	22555	5.0624	3.1068	1.7437	0.00000281	0.34217869	0.00003613
		0.005	3732.57	12438	3.3276	0.3730	15.0295	0.00000001	0.07669311	0.00000048
		0.01	1866.28	11558	3.8042	3.5571	1.2243	0.00000008	0.07669315	0.00000195
	All real	0.02	933.14	12910	20.0000	1.4490	0.5013	0.00000062	0.07669330	0.00000790
		0.005	3732.57	18369	0.8456	2.8471	7.7168	0.00000001	0.07669311	0.00000048
		0.01	1866.28	20284	3.3449	0.4663	10.5862	0.00000008	0.07669315	0.00000195
	m in both real/imaginary	0.02	933.14	19202	8.8356	1.0149	1.6182	0.00000062	0.07669330	0.00000790
		0.005	3732.57	18671	4.7688	3.6908	1.0604	0.00000001	0.07669312	0.00000048
		0.01	1866.28	19956	12.9099	0.1576	8.1536	0.00000008	0.07669315	0.00000195
G_3	*m* in real	0.02	933.14	21181	6.3207	1.9434	1.1803	0.00000062	0.07669330	0.00000790
		0.005	709.36	18978	5.8575	3.4244	1.5387	0.00000008	0.61576541	0.00000391
		0.01	354.68	12796	6.7181	0.3056	13.0071	0.00000063	0.61656335	0.00001588
	All real	0.02	177.34	18799	1.0387	4.2892	4.9060	0.00000505	0.61578624	0.00006465
		0.005	709.36	24058	6.3039	0.6956	7.0329	0.00000008	0.61576549	0.00000391
		0.01	354.68	16275	9.8957	4.8140	0.5624	0.00000062	0.61576941	0.00001585
	m in both real/imaginary	0.02	177.34	25842	7.0000	1.2469	2.5905	0.00000504	0.61578623	0.00006454
		0.005	709.36	12839	3.3809	4.0134	2.2719	0.00000008	0.61575900	0.00000391
		0.01	354.68	15186	2.2873	3.5658	3.2569	0.00000062	0.61578330	0.00001586
G_4	*m* in real	0.02	177.34	21984	7.4917	0.3108	9.6710	0.00000505	0.61578607	0.00006470

		0.005	811.96	18919	2.6098	1.8913	6.2400	0.00000004	0.30789366	0.00000196
		0.01	405.98	15850	12.8415	1.0794	1.9295	0.00000031	0.30789494	0.00000793
	All real	0.02	202.99	20098	8.2242	0.4209	6.5194	0.00000252	0.30790144	0.00003233
		0.005	811.96	14784	7.1645	0.9614	4.4820	0.00000004	0.30789353	0.00000196
		0.01	405.98	21709	5.2752	1.9267	2.6304	0.00000031	0.30789495	0.00000792
	m in both real/imaginary	0.02	202.99	35523	20.0000	0.1590	7.0610	0.00000240	0.30790064	0.00002915
		0.005	811.96	17457	8.2667	4.6064	0.8109	0.00000004	0.30789355	0.00000196
		0.01	405.98	13299	12.1915	0.1879	11.6652	0.00000031	0.30789086	0.00000793
G_5	*m* in real	0.02	202.99	36184	0.5846	1.8498	19.9857	0.00000251	0.30790139	0.00003206
		0.005	4629.90	17578	20.0000	5.0000	0.1217	0.00000001	0.09063826	0.00000057
		0.01	2314.95	23163	1.3519	1.1976	6.6332	0.00000009	0.09063827	0.00000229
	All real	0.02	1157.47	16486	0.6111	4.2722	3.5303	0.00000073	0.09063838	0.00000926
		0.005	4629.90	23994	1.5180	0.7173	11.1547	0.00000001	0.09063826	0.00000057
		0.01	2314.95	31087	8.1539	1.0958	1.2069	0.00000009	0.09063827	0.00000229
	m in both real/imaginary	0.02	1157.47	12567	5.0686	0.8179	2.2637	0.00000073	0.09063838	0.00000926
		0.005	4629.90	37689	0.9511	3.9768	3.2075	0.00000001	0.09063824	0.00000057
		0.01	2314.95	21977	6.6195	0.7853	2.0741	0.00000009	0.09063827	0.00000229
G_6	*m* in real	0.02	1157.47	17742	5.1773	3.9655	0.4574	0.00000073	0.09063838	0.00000926
		0.005	779.20	13742	16.2347	0.1869	8.1286	0.00000010	0.82971756	0.00000526
		0.01	389.60	22596	12.6640	0.2608	6.4837	0.00000084	0.82972223	0.00002126
	All real	0.02	194.80	15878	0.7641	1.7574	13.0506	0.00000678	0.82974191	0.00008652
		0.005	779.20	12570	18.8368	0.2807	4.6378	0.00000010	0.82960126	0.00000526
		0.01	389.60	15784	4.2810	0.9761	5.1117	0.00000084	0.82972229	0.00002126
	m in both real/imaginary	0.02	194.80	21736	8.8453	0.4895	4.1174	0.00000681	0.82973400	0.00008737
		0.005	779.20	24682	0.7150	3.0894	11.0591	0.00000010	0.82971764	0.00000526
		0.01	389.60	13912	14.7030	0.6743	2.1600	0.00000084	0.82968995	0.00002126
G_7	*m* in real	0.02	194.80	20291	7.7777	0.2307	10.0972	0.00000678	0.82974214	0.00008659
		0.005	3648.32	13774	20.0000	0.0631	2.9383	0.00000001	0.06241485	0.00000039
		0.01	1824.16	18373	2.5617	0.4622	2.7802	0.00000006	0.06249392	0.00000157
	All real	0.02	912.08	15043	0.5694	0.9216	5.4501	0.00000050	0.06249401	0.00000630
		0.005	3648.31	20999	0.1111	4.5444	7.2841	0.00000001	0.06249379	0.00000039
	m in both	0.01	1824.16	27725	0.2314	1.7977	7.8528	0.00000006	0.06249392	0.00000157
	real/imaginary	0.02	912.08	18930	8.9848	0.6853	0.4675	0.00000050	0.06249401	0.00000630
		0.005	3648.31	29461	17.2315	1.7137	0.1256	0.00000001	0.06249390	0.00000039
		0.01	1824.16	18791	0.4935	0.4585	14.4940	0.00000006	0.06249392	0.00000157
G_8	*m* in real	0.02	912.08	20825	4.6351	0.0604	10.2585	0.00000050	0.06249401	0.00000630
		0.005	890.81	16013	6.5852	1.0844	2.1079	0.00000004	0.32875691	0.00000207
		0.01	445.40	19699	0.2671	3.8142	12.3083	0.00000033	0.32875831	0.00000836
	All real	0.02	222.70	16802	2.6349	4.0728	1.0140	0.00000267	0.32876409	0.00003392
		0.005	890.81	15784	5.7626	0.9134	2.8594	0.00000004	0.32875764	0.00000207
	m in both	0.01	445.40	20623	4.1118	0.7240	4.3556	0.00000033	0.32875828	0.00000836
	real/imaginary	0.02	222.70	21801	2.0871	2.4744	2.1067	0.00000267	0.32876388	0.00003390
		0.005	890.81	26222	0.1474	5.0000	19.9211	0.00000004	0.32875696	0.00000207
		0.01	445.40	15420	14.1554	0.1617	5.6758	0.00000033	0.32875828	0.00000836
G_9	*m* in real	0.02	222.70	17760	0.9547	0.7859	14.2970	0.00000267	0.32876428	0.00003394

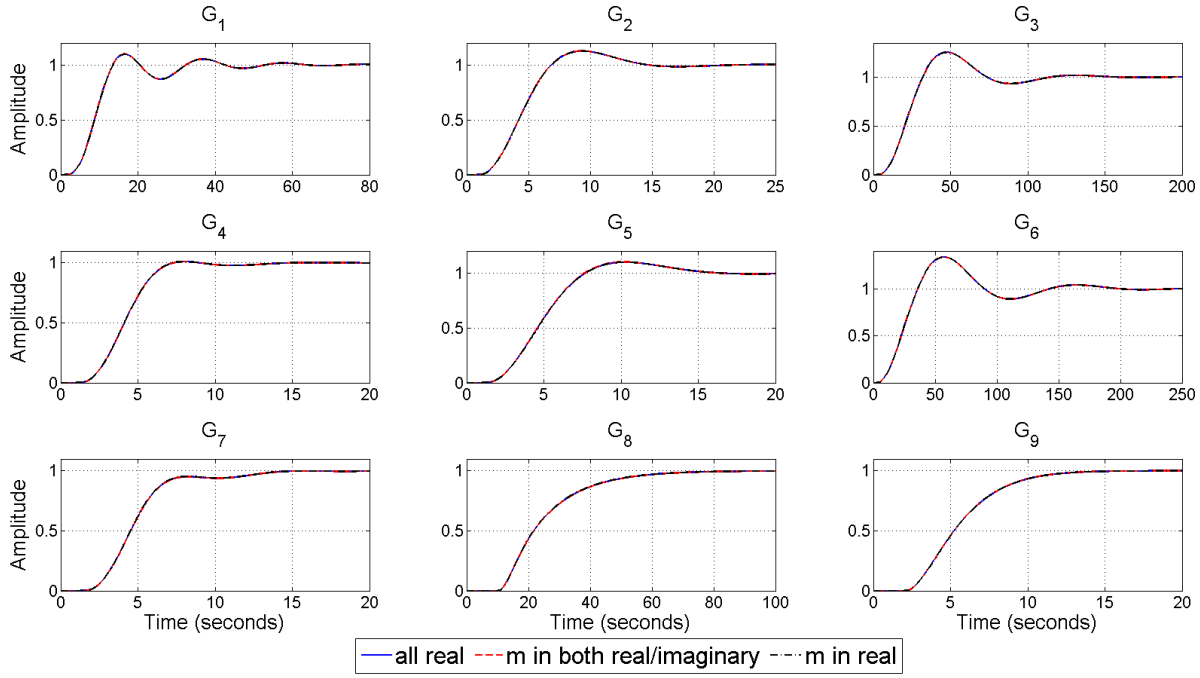

Figure 5: Controlled variable (y) due to step input in set-point for the nine test-bench plants.

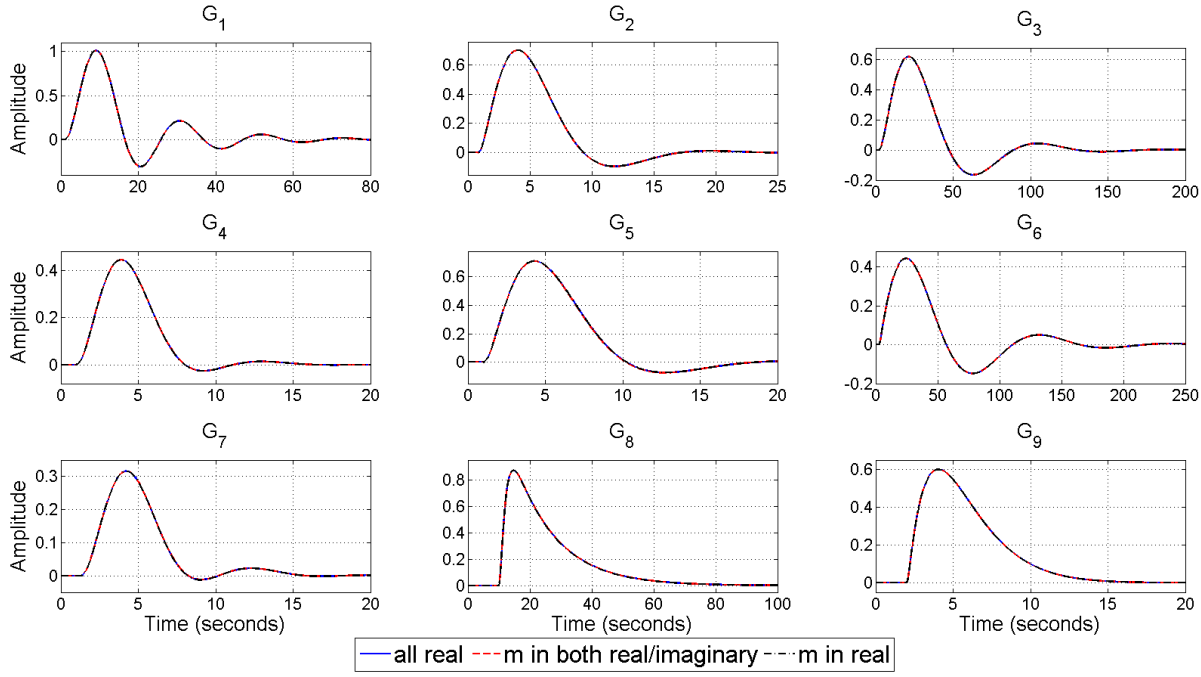

Figure 6: Controlled variable (y) due to step input in disturbance input for the nine test-bench plants.

5.2 Control performance of the test-bench plants

Although, our proposed method considers the dominant pole placement based PID controller design using co-efficient matching approach in discrete time with specified T_s to handle the time delay of SOPTD systems, but the obtained discrete time PID controllers are mapped back to its continuous time version for handling the continuous time SOPTD systems. The corresponding time responses of the controlled variables (y) and manipulated variables (u) due to a step change in set-point and disturbance inputs are shown in Figure 5-Figure 6 and Figure 7-Figure 8 respectively, with the PID controller gains obtained from three different non-dominant pole-types given in Table 2 for the nine test-bench plants. All the time-domain responses remain almost unchanged when the natures of the non-dominant poles are of above three types as shown in Figure 5-Figure 8. The tracking responses are found to be

oscillatory and sluggish in nature for small and large L/τ ratios respectively which are shown in Figure 5 and Figure 7. The corresponding time responses of disturbance rejection in Figure 6 and Figure 8 are oscillatory for all the three types of plants. The three different set of PID controller gains $\{K_p, K_i, K_d\}$ are obtained from the respective cases of non-dominant pole types for case 1 to case 3 i.e. all real non-dominant pole and two different complex conjugate non-dominant poles respectively and their differences can be quantified from Table 2. However, it is found from Table 2 that the deviations of all the three different PID controller gains are negligibly small for all the three non-dominant pole types and nine test-bench plants which support the negligible difference in the corresponding time responses. However, it is to be noted the time responses of each plants are almost same with three different controllers while the corresponding stability regions are quite different to each other.

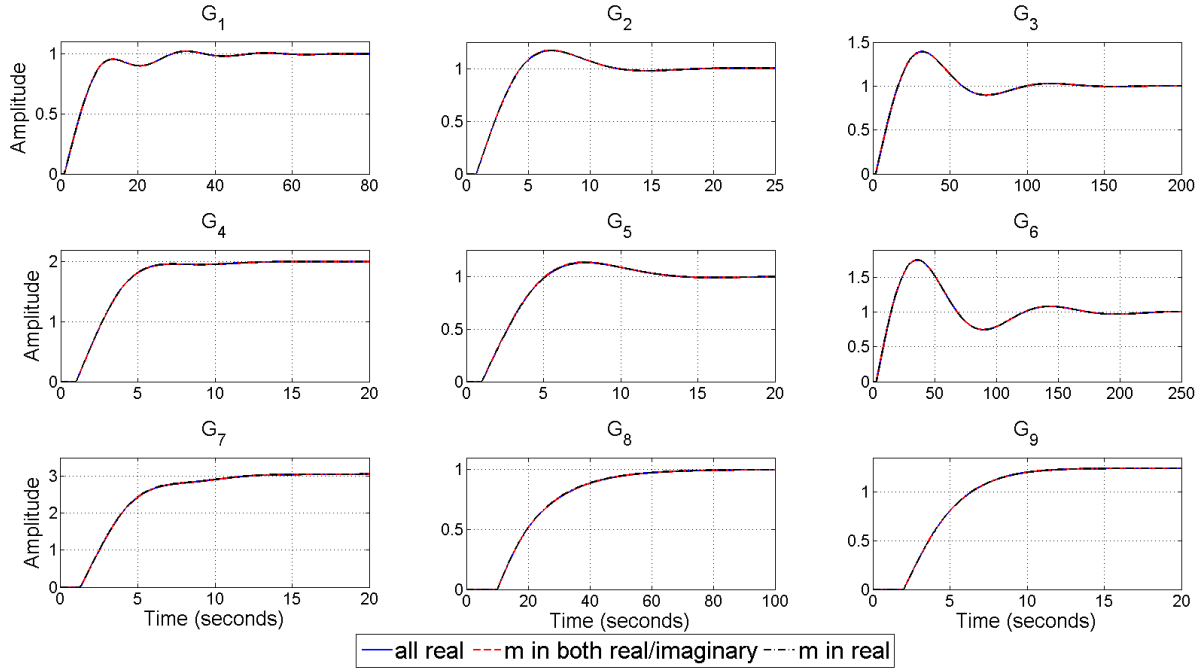

Figure 7: Manipulated variable (u) due to step input in set-point for the nine test-bench plants.

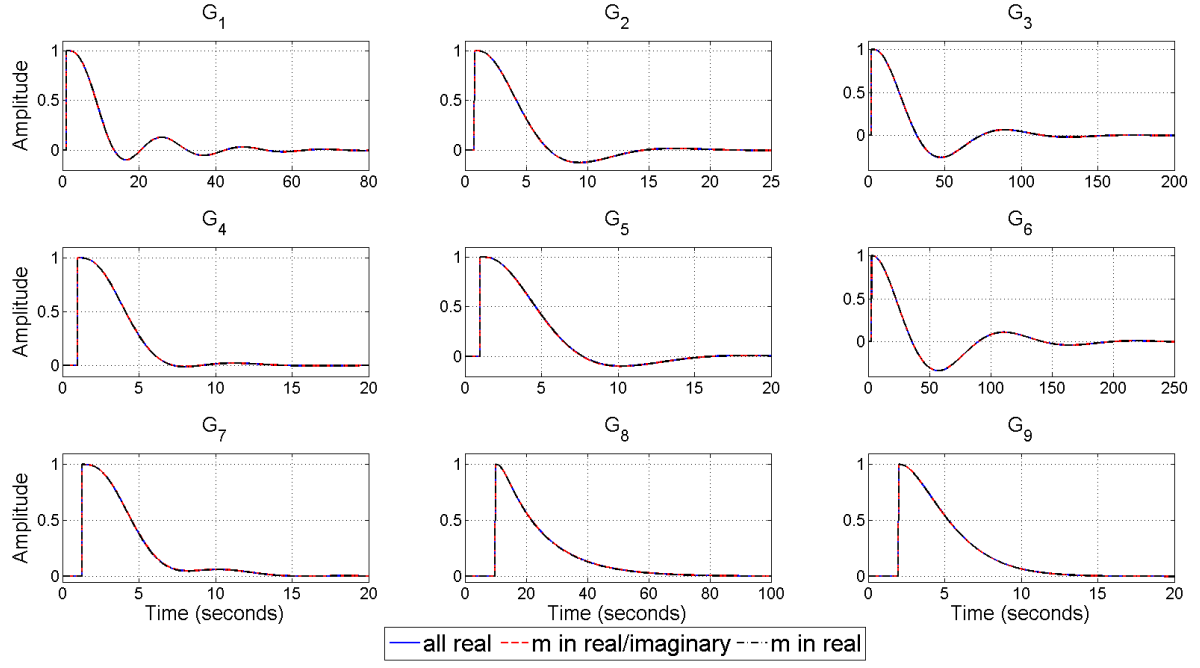

Figure 8: Manipulated variable (u) due to step input in disturbance input for the nine test-bench plants.

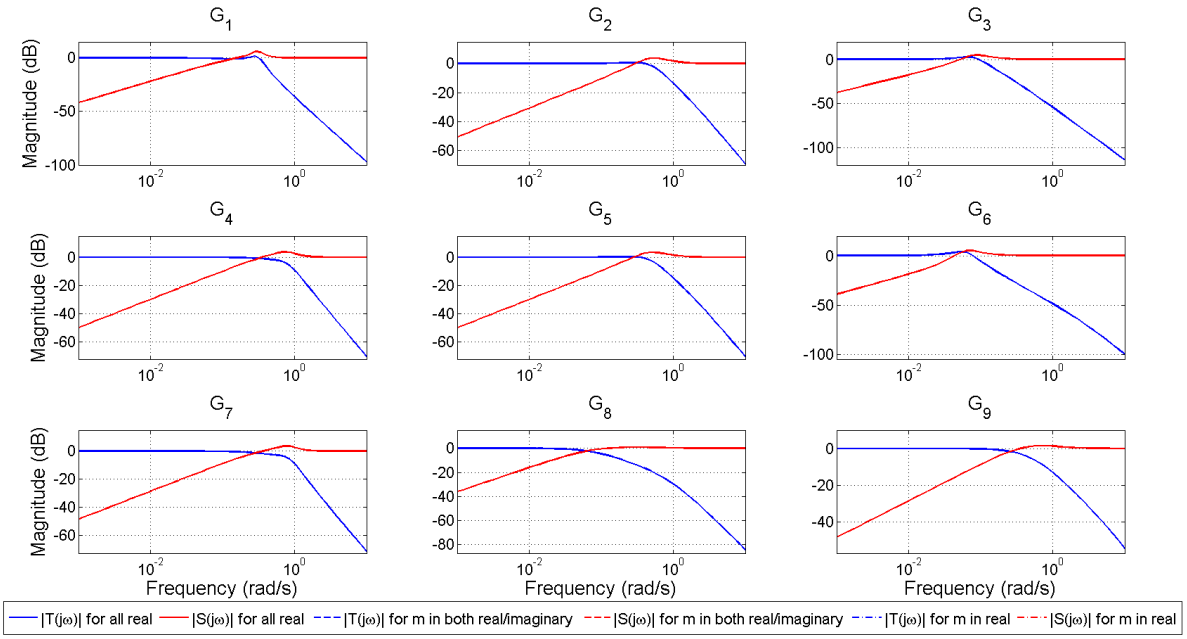

Figure 9: Sensitivity and complementary sensitivity functions with various non-dominant pole types for the nine test-bench plants.

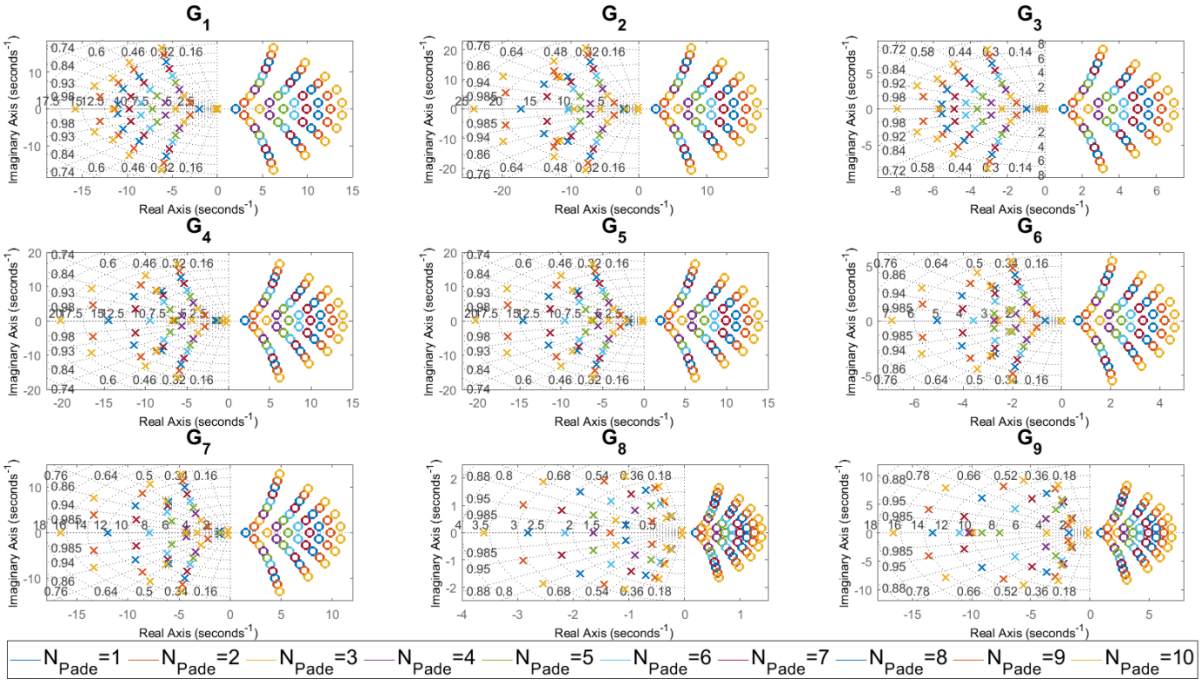

Figure 10: Closed loop pole/zero locations for the test-bench plants and PID controller with all complex conjugate non-dominant pole type (*m* in both real and imaginary parts) while increasing the order of Pade approximation.

Figure 9 shows the magnitude plots of sensitivity $\left|S_e(j\omega)\right|$ and complementary sensitivity $\left|T(j\omega)\right|$ trade-offs (Das et al. 2018; Das & Pan 2014). Similar to the time-responses, the sensitivity and complementary sensitivity remain almost unchanged with three different non-dominant pole types for the nine test-bench plants, as expected. The sensitivity and co-sensitivity have high-pass and low-pass characteristics respectively and indicates good disturbance and noise rejection respectively.

The closed loop pole-zero locations are shown in Figure 10 where the order of Pade approximation (N_{Pade}) has been increased from 1 to 10, using the second case for the PID expressions. Here, showing

a single case is sufficient since the difference in the controller gains are found to be negligibly small, although the optimum design parameters are different Table 2. Also, since the time delay gets manifested in terms of many interlaced poles and zeros, the relative location of the non-dominant poles, compared to the dominant poles are particularly important in this context.

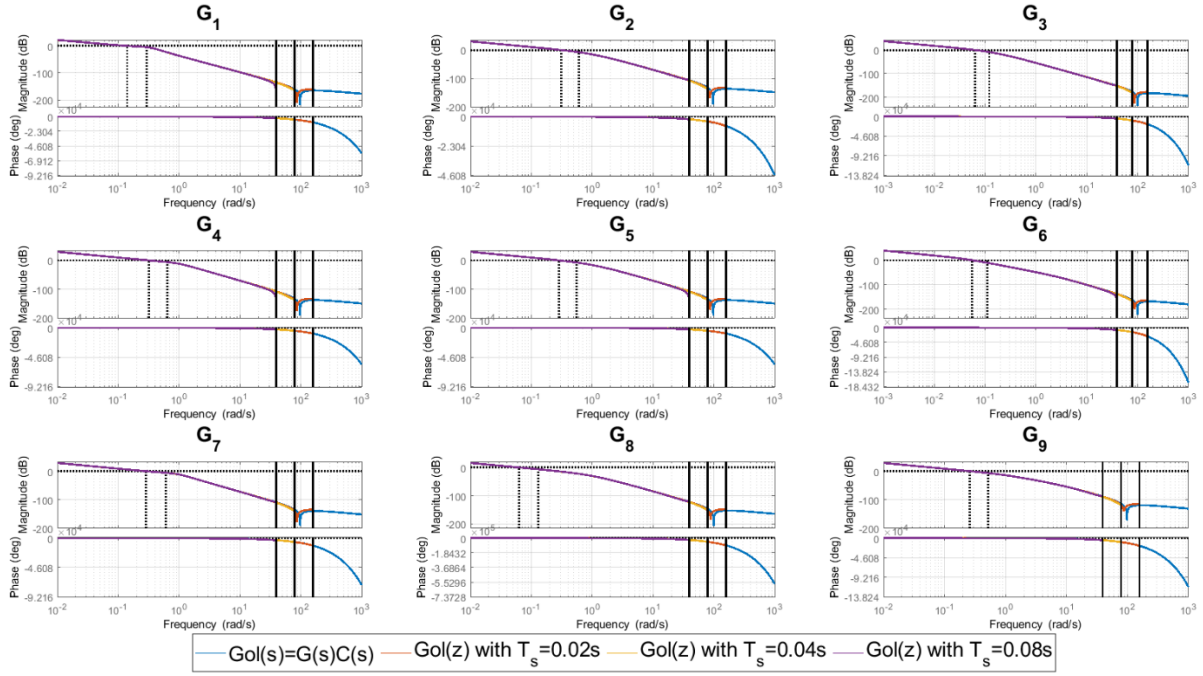

Figure 11: Frequency response of continuous and discrete time open loop system (G_{ol}) for nine test bench plants using the PID controller obtained from all real non-dominant pole type.

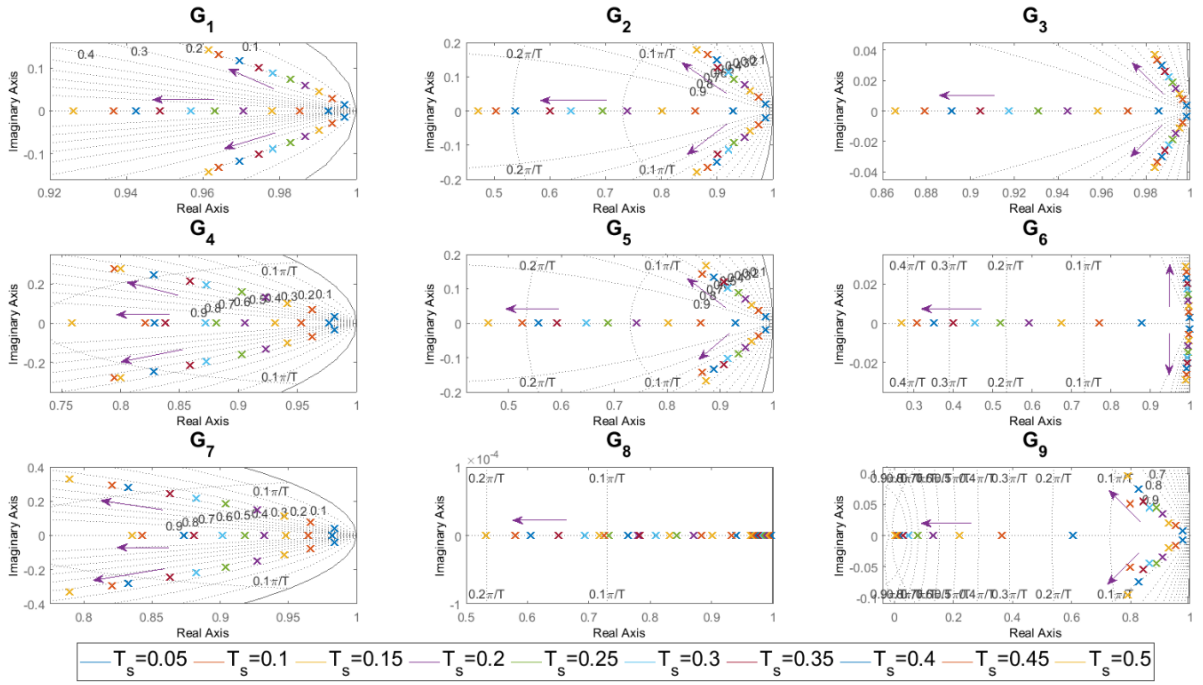

Figure 12: Closed loop pole location in discrete time with varying sampling time (T_s) for nine test-bench plants.

We have now presented a comparison of the frequency responses between the continuous and discrete time versions of the control system as shown in Figure 11-Figure 13. This includes variation of the sampling time and investigation of its effect on the open loop Bode plots, root locus and closed loop pole locations. Figure 11-Figure 13 shows the frequency response of the open loop plants with the

controller designed at T_s = 0.02 sec. It is seen from Figure 11 when we increase the value of T_s the bandwidth is reduced and the Nyquist frequency shifts towards the low frequency regions (Das et al. 2011), while a drooping phase is observed for towards higher frequencies with smaller T_s and overlaps with the continuous time case for wider range of frequencies. Also, for smaller sampling time, the gain is found to be close to the continuous time case. Figure 12 demonstrates the closed lope location of each plants with the variation of T_s. In Figure 12, it can be observed that the closed loop poles are shifted towards the higher frequencies and lower damping regions with gradual increase in the sampling time shown by the arrows which indicates with the increasing sampling time the stability is decreasing while the controller was designed at a fixed sampling time i.e. T_s = 0.02 sec. This should not be confused with designing the control system with different sampling times at the first instant.

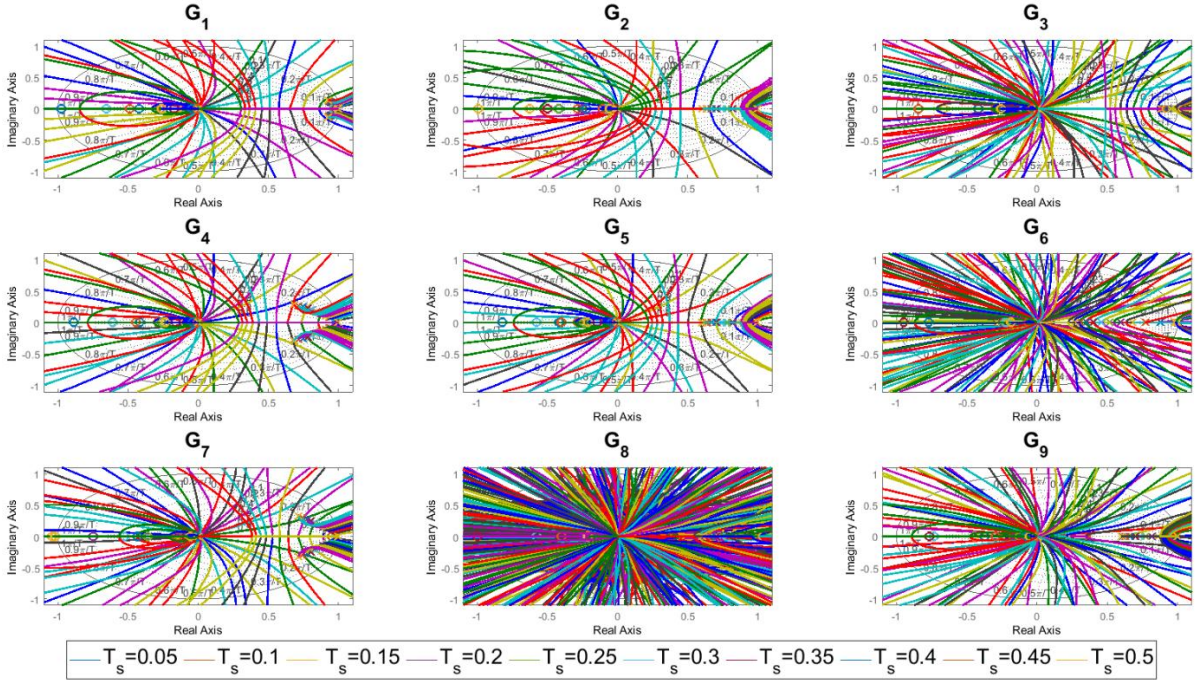

Figure 13: Root locus plot of discrete open loop systems $G_{ol}(z)$ with varying sampling time for nine test-bench plants.

From the root locus plot in Figure 13, it is seen that the asymptotes are spanned from the poles to the zeros which are crossing the unit circle faster with higher sampling time since zeros are outside the unit circle due to Pade approximation. Therefore, it is necessary for ensuring the stability of time delay systems, the controller gains need to be small as reported in Table 2.

5.3 Convergence characteristics

For the three non-dominant pole types and nine test-bench plants, the distribution of sampled data-points or the stability regions can also be represented as the ISE values in terms of joint distributions of the parameter-pairs and are shown in Figure 14-Figure 16, where the colour-bars represent the ISE values. The evolution of the design parameters and controller parameters along with the ISE can also be visualised in terms of the sampled points over the iterations where they converge to different values for the design parameters $\{m, \zeta_{cl}, \omega_{cl}\}$ but to similar values in the controller parameter space $\{K_p, K_i, K_d\}$ which are evident from Figure 17-Figure 19 and Figure 20-Figure 22 respectively.

It is clearly visible from the Figure 17-Figure 19 that the closed loop desired parameters $\{m, \zeta_{cl}, \omega_{cl}\}$ are converging with the increasing number of iteration for different sampling time. Whereas the controller parameters $\{K_p, K_i, K_d\}$ obtained from three different PID controller expressions are merging with each other (at the dense locations) for different sampling time as shown in Figure 20-Figure 22.

Convergence of corresponding ISE values are shown in Figure 23-Figure 24 with the number of iterations. Figure 23 represents the convergence of ISE values obtained from random search and optimization with different sampling time while Figure 24 depicts the corresponding convergence of best ISE values in log-log plot derived from Figure 23. Such convergence characteristics have been previous shown in the PID controller parameter space exploration as the global optimization algorithm converges in (Tang et al. 2012).

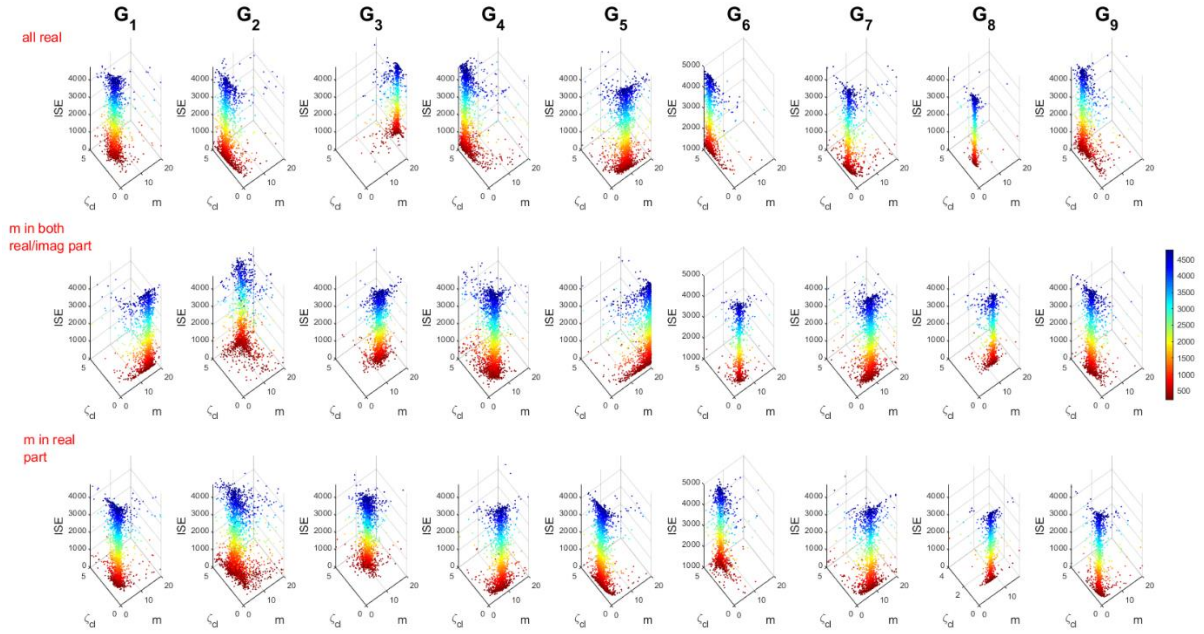

Figure 14: Distributions of ISE as functions of $\{m, \zeta_{cl}\}$.

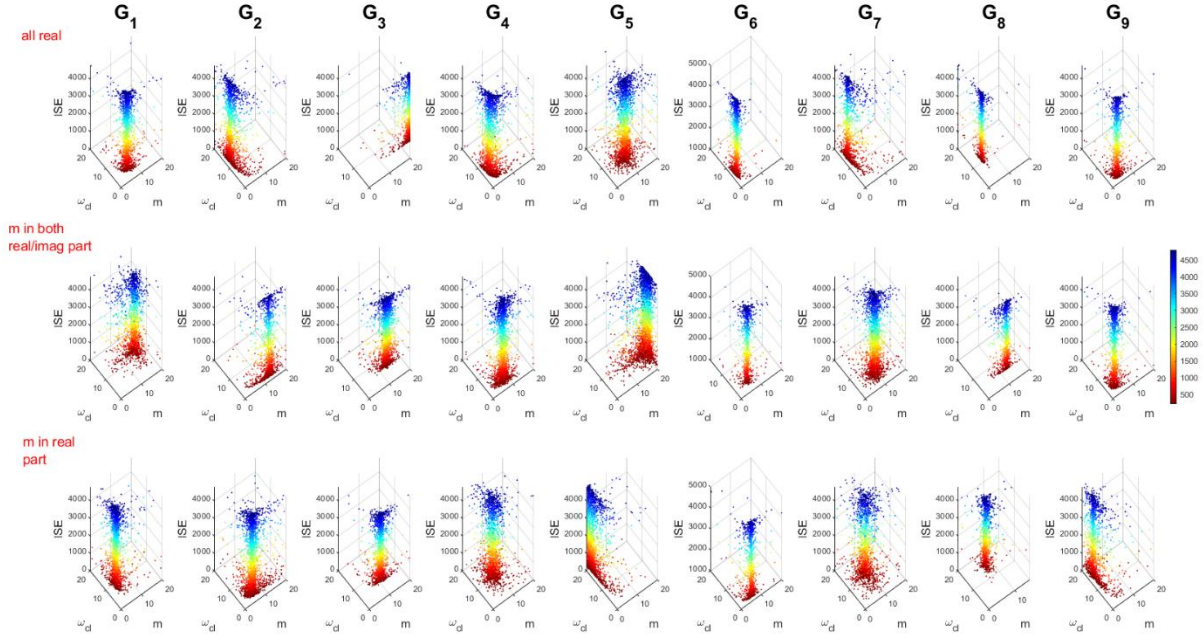

Figure 15: Distributions of ISE as functions of $\{m, \omega_{cl}\}$.

All the simulations were run on a 64-bit Windows PC with 64GB memory and an AMD Ryzen 7, 3.6 GHz processor where simulations were run on a single core to stack all visited stabilizable datapoints in an archive. The simulation run time of the random search and optimization based sampling with the T_s drawn within the chosen interval of the design parameters are compared in Figure 25 as box plot for nine test-bench plants with three different non-dominant pole types. It is observed from Figure 25 that for both the complex conjugate non-dominant pole types the simulation time for convergence of PSO algorithm is decreasing with the increasing T_s since the median values of each box are decreasing.

Whereas, simulation time for converging the PSO algorithm is higher with the T_s = 0.01 sec compared to T_s = 0.005 sec and T_s = 0.02 sec when all real nondominant pole type is considered. However, the simulation time for converging the PSO algorithm is faster with the T_s = 0.02 for all three different non-dominant pole types which justifies our choice of sampling time.

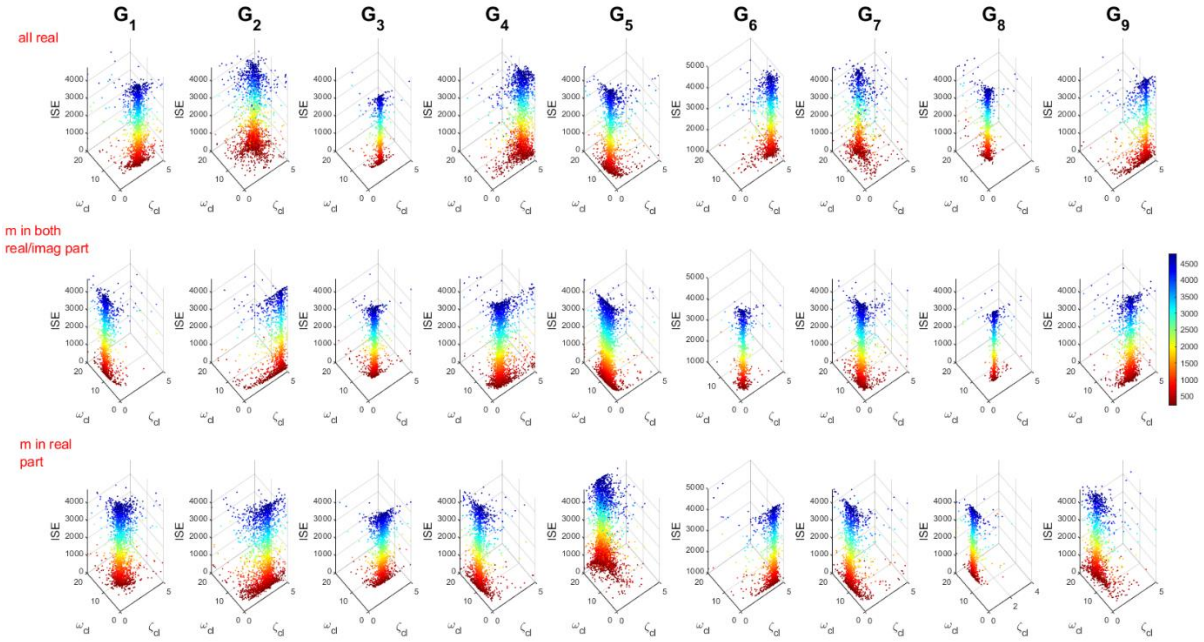

Figure 16: Distributions of ISE as functions of $\{\zeta_{cl}, \omega_{cl}\}$.

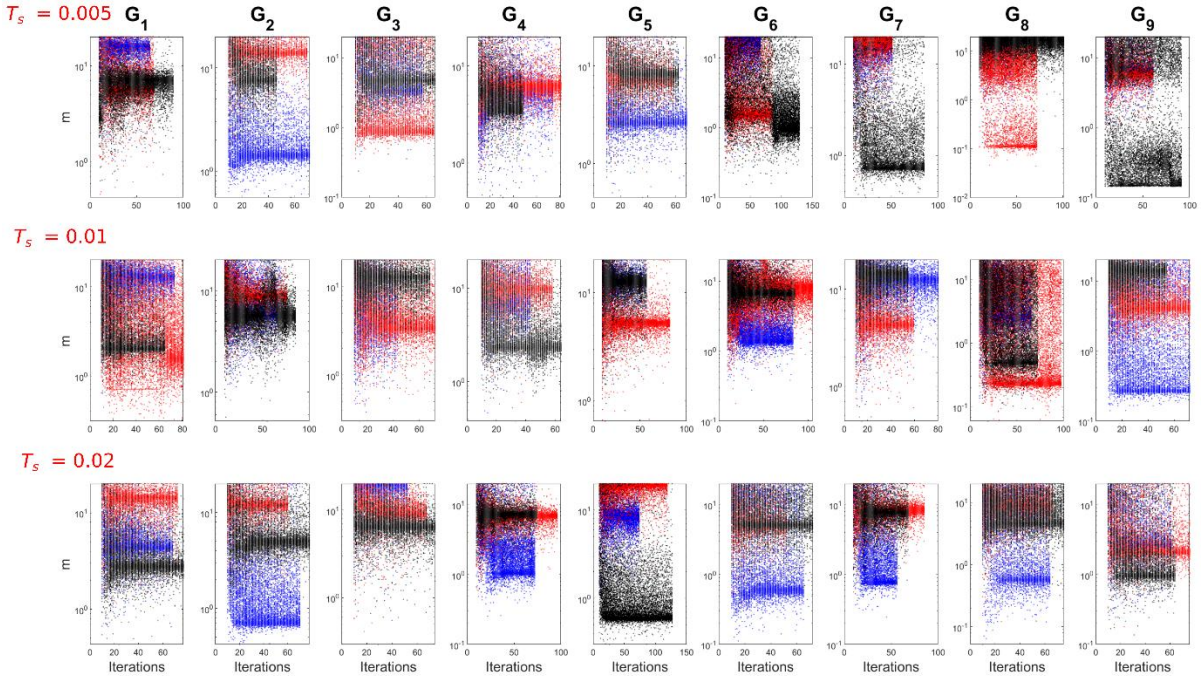

Figure 17: Convergence of pole placement parameter (m) in semilog y-scale for the three pole types: (blue): all real, (red): m in both in both real/imaginary part, (black): m in real part.

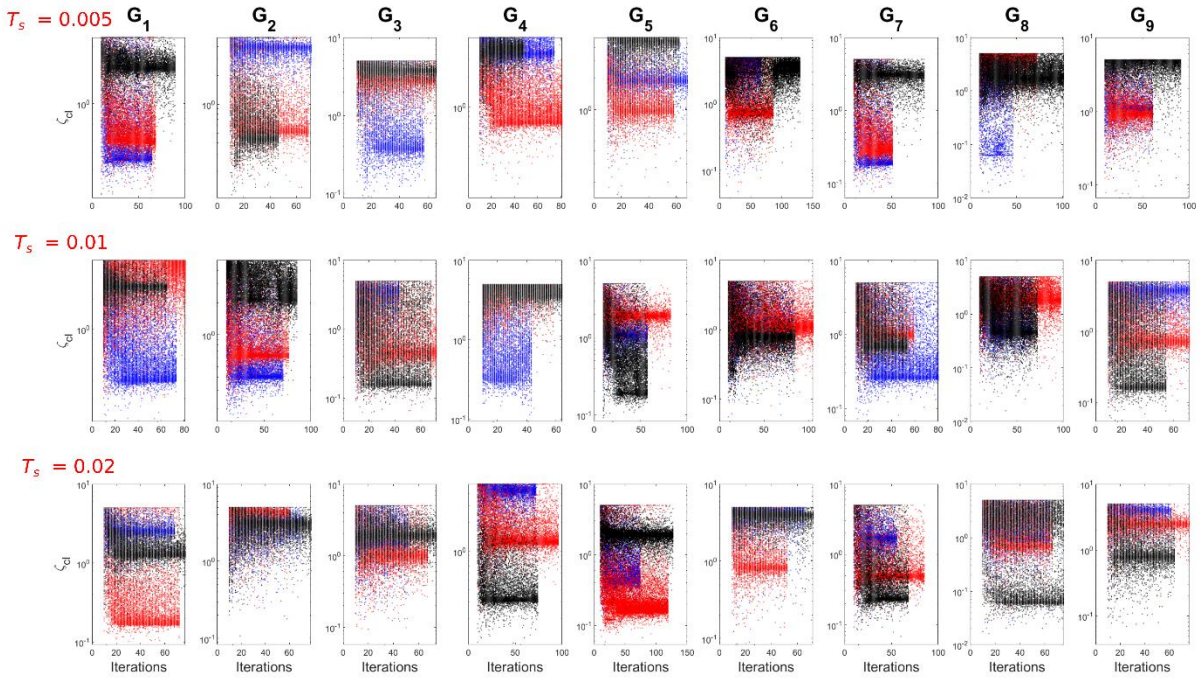

Figure 18: Convergence of the desired closed loop damping (ζ_{cl}) in semilog *y*-scale for the three pole types: (blue): all real, (red): *m* in both in both real/imaginary part, (black): *m* in real part.

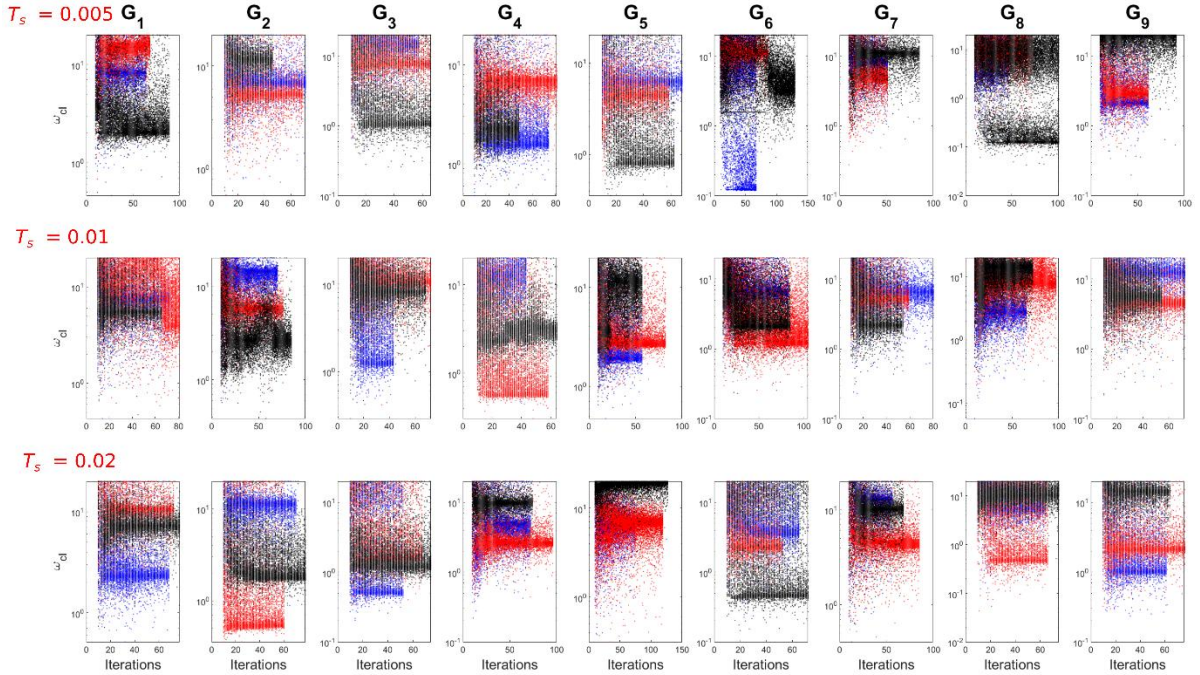

Figure 19: Convergence of the desired closed loop frequency (ω_{cl}) in semilog *y*-scale for the three pole types: (blue): all real, (red): *m* in both in both real/imaginary part, (black): *m* in real part.

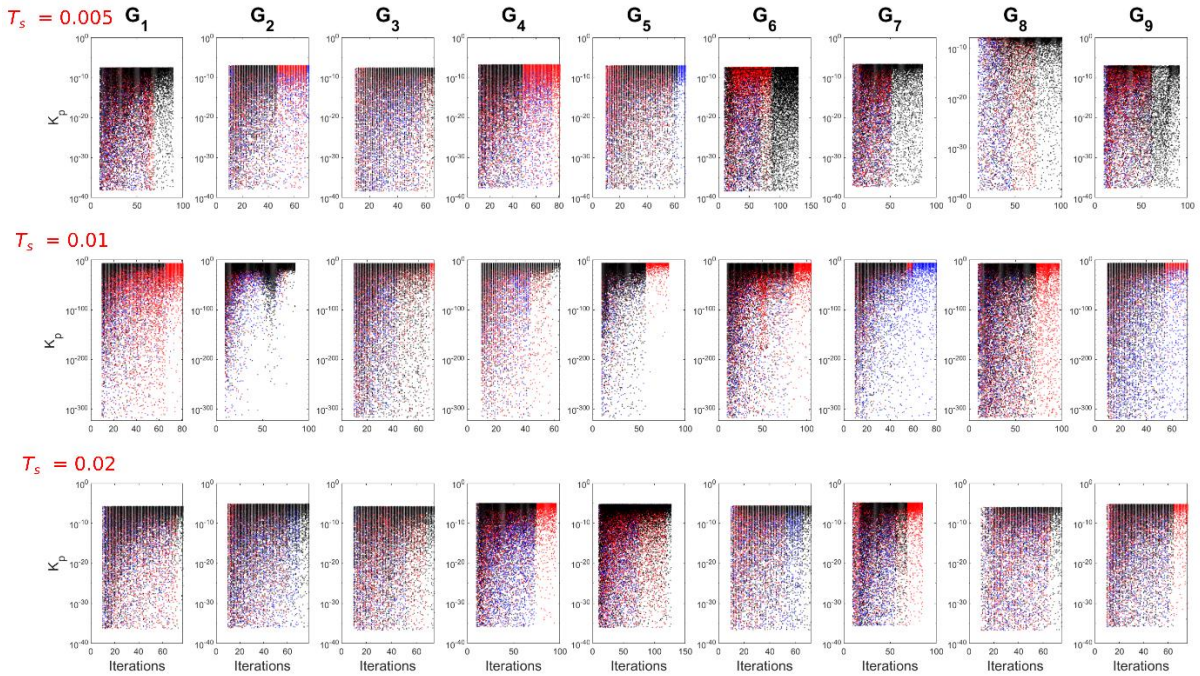

Figure 20: Convergence of the proportional gain (K_p) in semilog *y*-scale for the three pole types: (blue): all real, (red): *m* in both in both real/imaginary part, (black): *m* in real part.

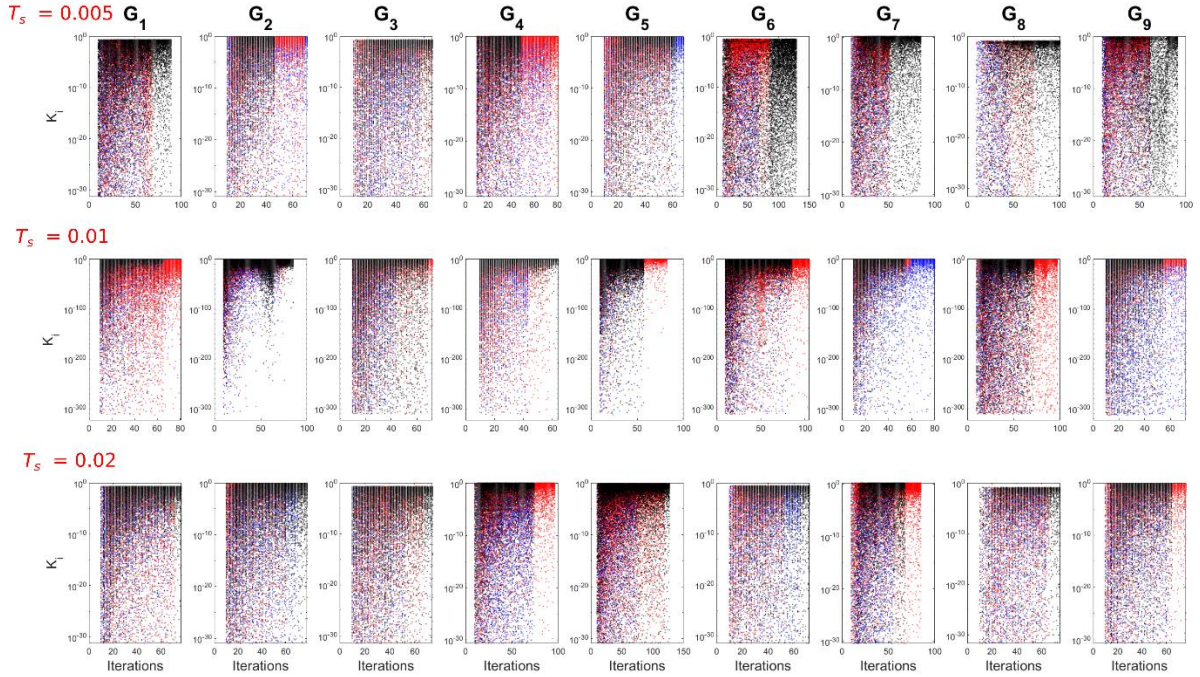

Figure 21: Convergence of the integral gain (K_i) in semilog *y*-scale for the three pole types: (blue): all real, (red): *m* in both in both real/imaginary part, (black): *m* in real part.

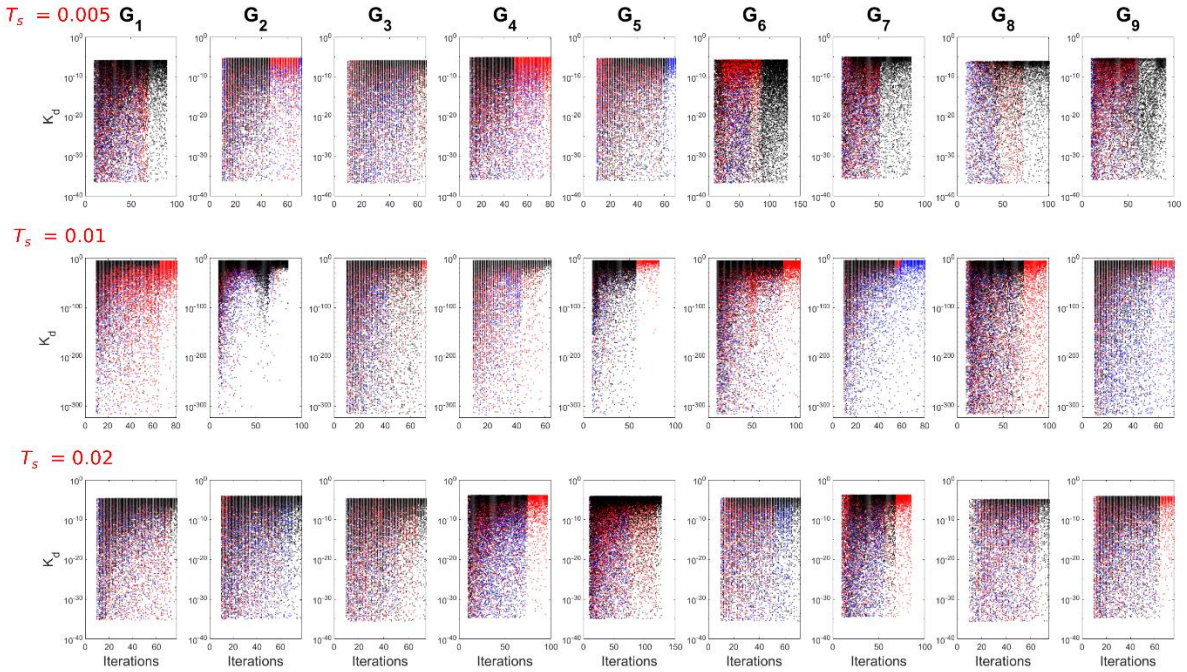

Figure 22: Convergence of the derivative gain (K_d) in semilog *y*-scale for the three pole types: (blue): all real, (red): *m* in both in both real/imaginary part, (black): *m* in real part.

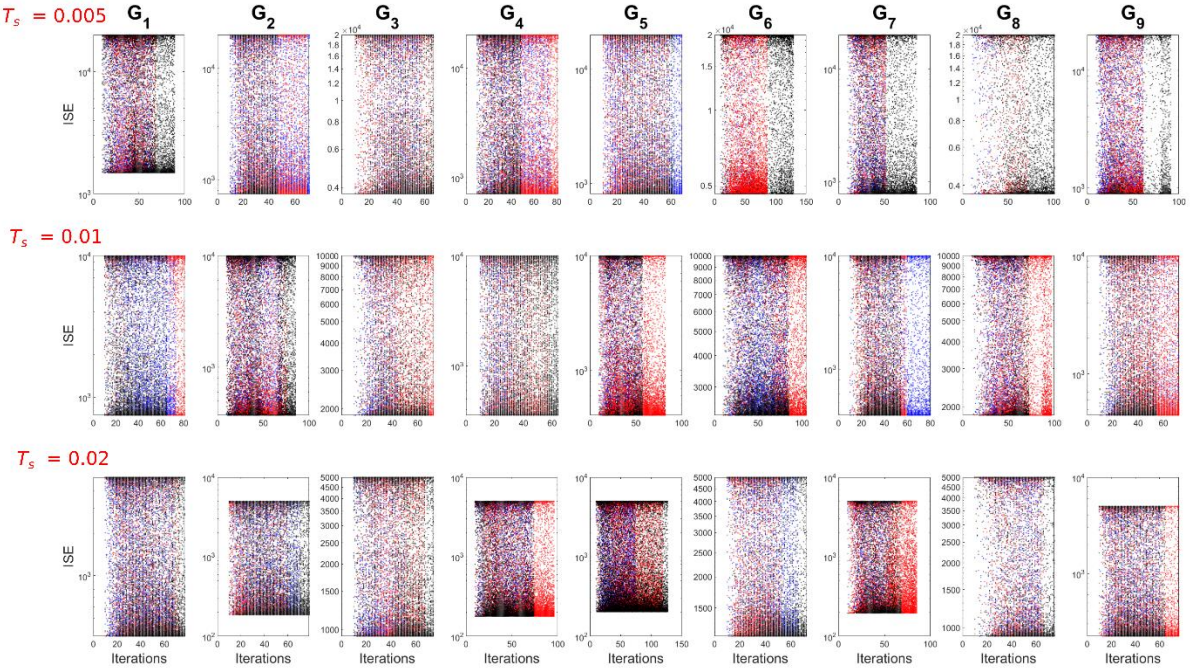

Figure 23: Convergence of the ISE criterion in semilog *y*-scale for the three pole types: (blue): all real, (red): *m* in both in both real/imaginary part, (black): *m* in real part.

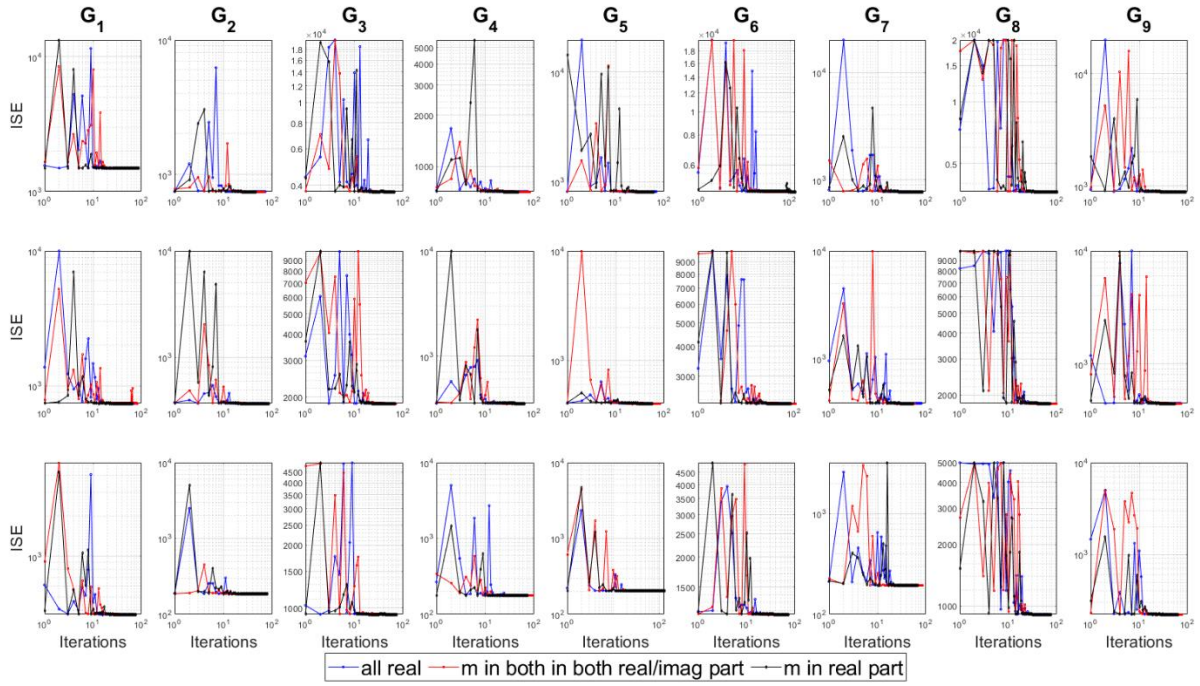

Figure 24: Convergence of the best ISE with the PSO iterations in log-log scale.

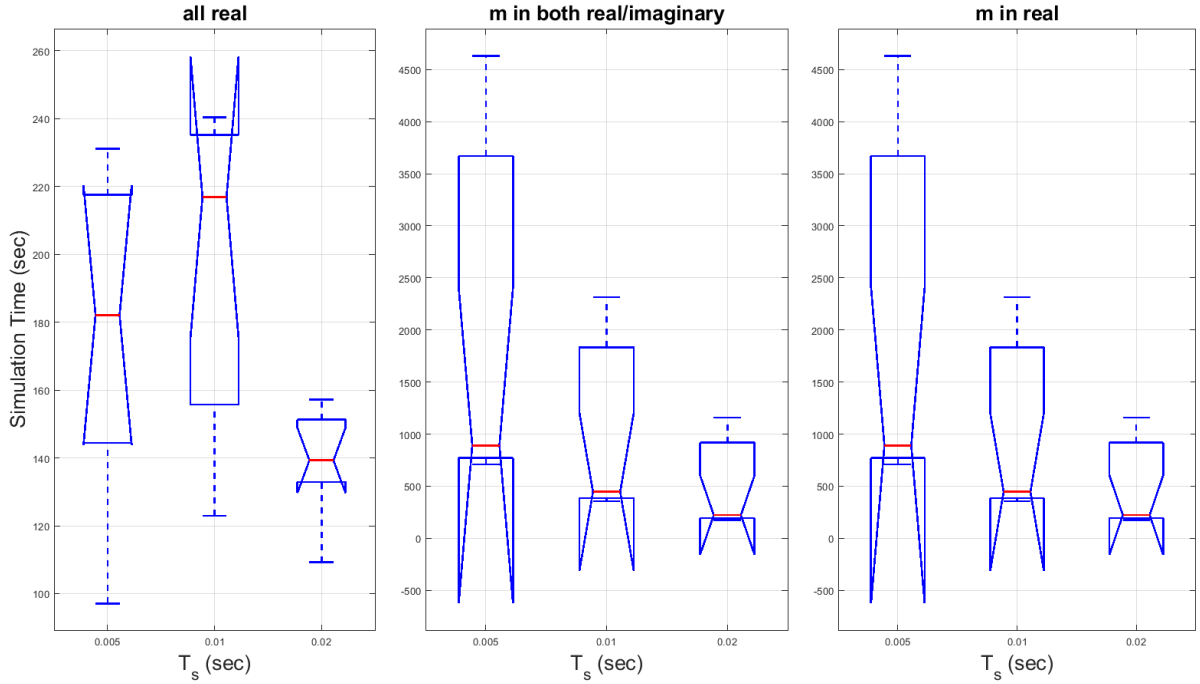

Figure 25: Simulation time for convergence of the PSO algorithm with different sampling time (T_s) for nine test bench plants using three different PID controllers obtained from three different non-dominant pole types. Each boxplot shows the statistics of nine plants.

5.4 Discussions

It is well known fact that several optimization techniques have been used in the past e.g. in (Herreros et al. 2002)(Das & Pan 2014)(Pan & Das 2013) to determine the PID controller gains for optimal/robust control design while minimizing or maximizing chosen objective functions of the control performance. However, these optimization methods for determining the PID controller parameters do not provide any option of choosing the dominant pole placement parameters which are directly related to the closed loop response. In this paper, as an alternative approach to the stability boundary approach (Silva et al. 2007), we have used a random search and optimization technique using PSO algorithm to determine the three-desired closed-loop performance parameters $\{m, \zeta_{cl}, \omega_{cl}\}$ in the first step and then derive the

PID controller parameters using the proposed analytical expressions which ensures dominant pole placement for SOPTD systems while choosing different types of non-dominant poles. However, the derivations of PID controller expressions for dominant pole placement of SOPTD process models without any approximating the delay term is one of the main contributions of this paper.

Similar to the method proposed in (Das et al. 2018), this paper also uses three different PID controller expressions from three different non-dominant pole types to obtain the stability regions in the closed loop parameter space $\{m, \zeta_{cl}, \omega_{cl}\}$ and then the PID controller gain space. However, the delay terms here are mapped as finite number of poles in the *z*-domain rather being approximated by fixed order Pade. This is due to the fact that earlier literature (Bhattacharyya et al. 2009; Silva et al. 2007) have described that the stability region of time delay systems under PID control may shrink with increasing order of Pade approximation (Bhattacharyya et al. 2009; Silva et al. 2007) although the time response simulations may be very close to the original transcendental cases with the delay. Moreover, the Pade approximation of time delay terms increases the number of both zeros and poles of the open loop time delay system while the proposed discretization method will map the delay terms into only a finite number of poles of the open loop system i.e. $e^{-Ls} = z^{-n}$ where $n = L/T_s, n \in \mathbb{Z}_+$, depending on the choice of the sampling time (T_s). Therefore, the present method is capable of avoiding the appearance of zeros due to Pade approximation beside the poles, since the closed loop stability of the system with and without zeros can be very different. So, the current method has an advantage of less approximation going into the implementation of the time delays and the side effect of including unnecessary changes of the closed loop stability regions due to Pade approximations.

Also, this paper does not specify the desired closed loop parameters $\{m, \zeta_{cl}, \omega_{cl}\}$ similar to previous works in (Wang et al. 2009; Wang et al. 2007)(Halder et al. 2020), rather here we have selected the optimum closed loop parameters $\{m, \zeta_{cl}, \omega_{cl}\}$ by minimizing the integral of squared error (ISE) criterion using the PSO algorithm. Moreover, minimization of ISE ensures the percentage overshoot, settling time, rise time etc. which are the commonly used time response measures, are already taken care of. Therefore, we do not need to specify the desired closed loop parameters $\{m, \zeta_{cl}, \omega_{cl}\}$ manually.

Previously, the method proposed in (Das et al. 2018) shows a dominant pole placement based PID controller design method to handle SOPTD process models by using a 3rd order Pade approximation of the delay term to avoid the high dimensional problem. The method in (Das et al. 2018) was validated for the SOPTD systems by varying the order of Pade approximation for the delay where the PID controller derivations were done with an assumption of 3rd order Pade for the delay terms. However, it is well-known that the overall system's order and dynamics of the SOPTD system will depend on the order of the Pade approximation. Therefore, the mathematical method in (Das et al. 2018) does not necessarily generalize the dominant pole placement based PID controller design for SOPTD process models beyond the third order Pade approximation. The proposed methodology in this paper does not consider any such approximation like different order of Pade and thus generalizes the dominant pole placement based PID controller design for handling SOPTD systems. However, for simulation purposes, we have also shown the time and frequency responses using different order of Pade approximation for highlighting the credibility of the method over existing literature. Recently, a similar dominant pole placement method based PID controller design has been proposed in (Das et al. 2020) where the Euler's discretization formula was used for the PID controller. This paper differs in the methodology as compared to (Das et al. 2020) since it uses a more accurate – Tustin's discretization formula for the PID controller. As a result, the mathematical derivations and the resulting stability regions explored by the global optimizer are very different. It is also evident that the present results using Tustin's discretization for the controller are mostly insensitive to the choice of non-dominant pole types as evident from the time responses. On contrary, previous work (Das et al. 2020) considering an Euler's discretization for the PID controller reported this choice of non-dominant pole type does matter in obtaining a satisfactory time and frequency response under similar design concepts.

Here, a thorough investigation has also been reported on the selection of sampling time based on the minimum ISE criterion while searching for the best controller parameters. This is intrinsically different

approach compared to the design of a control system with a fixed sampling time and then varying it as explored in (Halder et al. 2020). Whereas in the present approach it has been shown that the control performances, stability regions, convergence characteristics are affected in a great extent by the selection of the sampling time which are shown in Table 2 and Figure 14-Figure 24.

6. Conclusion

In this paper, dominant pole placement based discrete time PID controller has been designed for controlling SOPTD plants. Three different transformations of PID controller gains have been obtained with three different non-dominant pole types. Using these different PID controller expressions, PSO based random search and optimization method has been used to draw samples from the stability region in the dominant pole placement and controller parameter space by minimizing the ISE criterion for different sampling times. The obtained PID controller gains are very close to each other from these three different expressions for a fixed sampling time although their corresponding stability regions may be different. In order to compare the control loop performances, the sampling time is selected based on the minimum ISE criterion, obtained from the PSO based random search and optimization algorithm for all the nine test-bench plants with three different non-dominant pole types. We found that the control performances remain almost unaltered for each plant when the PID controller gains are obtained from three alternative expressions of non-dominant pole types. The proposed discrete-time transformation of continuous time PID controllers using pole-zero matching method can be extended in the future to design dominant pole-placement based PID controllers for higher order plants. Also, the choice of sampling time in the proposed approach should be kept at a level where $n = L/T_s$ is as low as possible for restricting the resulting polynomial having a reasonable degree which may be explored in a future research. Disturbance rejection based sampling approach can also be explored in the future instead of the current set-point tracking criterion for obtaining the stabilizable set of PID controller gains and for other type of plants under control e.g. with lead processes and non-minimum phase zeros.

Acknowledgement:

SD was partially supported by the ESIF ERDF Cornwall New Energy (CNE), Project number 05R16P00282.

References

Åström, K.J. & Hägglund, T., 2006. *Advanced PID control*, ISA-The Instrumentation, Systems and Automation Society.

Bhattacharyya, S.P., Datta, A. & Keel, L.H., 2009. *Linear control theory: structure, robustness, and optimization*, CRC press.

Das, S. et al., 2012. Inverse optimal control formulation for guaranteed dominant pole placement with PI/PID controllers. In *Computer Communication and Informatics (ICCCI), 2012 International Conference on*. pp. 1–6.

Das, S. et al., 2011. Optimizing continued fraction expansion based IIR realization of fractional order differ-integrators with genetic algorithm. In *2011 International Conference on Process Automation, Control and Computing*. pp. 1–6.

Das, S., Halder, K. & Gupta, A., 2020. Delay Handling Method in Dominant Pole Placement based PID Controller Design. *IEEE Transactions on Industrial Informatics*, 16(2), pp.980–991.

Das, S., Halder, K. & Gupta, A., 2018. Performance analysis of robust stable PID controllers using dominant pole placement for SOPTD process models. *Knowledge-Based Systems*, 146, pp.12–43.

Das, S. & Pan, I., 2014. On the Mixed H2/H∞ Loop-Shaping Tradeoffs in Fractional-Order Control of the AVR System. *IEEE Transactions on Industrial Informatics*, 10(4), pp.1982–1991.

Dincel, E. & Söylemez, M.T., 2014. Guaranteed dominant pole placement with discrete-PID controllers: a modified Nyquist plot approach. *IFAC Proceedings Volumes*, 47(3), pp.3122–3127.

Doyle, J.C., Francis, B.A. & Tannenbaum, A.R., 2013. *Feedback control theory*, Courier Corporation.

Ergenc, A.F., 2012. A matrix technique for dominant pole placement of discrete-time time-delayed systems. *IFAC Proceedings Volumes*, 45(14), pp.167–172.

Franklin, G.F., Powell, J.D. & Workman, M.L., 1998. *Digital control of dynamic systems*, Addison-wesley Menlo Park.

Halder, K., Das, S. & Gupta, A., 2020. Transformation of LQR Weights for Discretization Invariant Performance of PI/PID Dominant Pole Placement Controllers. *Robotica*, 38(2), pp.271–298.
Hernández-Diez, J.-E. et al., 2018. Proportional-delayed controllers design for LTI-systems: a geometric approach. *International Journal of Control*, 91(4), pp.907–925.
Hernández-Diez, J.-E., Méndez-Barrios, C.-F. & Niculescu, S.-I., 2019. Practical Guidelines for Tuning PD and PI Delay-Based Controllers. *IFAC-PapersOnLine*, 52(18), pp.61–66.
Herreros, A., Baeyens, E. & Perán, J.R., 2002. Design of PID-type controllers using multiobjective genetic algorithms. *ISA Transactions*, 41(4), pp.457–472.
Kiong, T.K., Wang, Q.G. & Chang, C.H., 1999. *Advances in PID control*, Springer.
Méndez-Barrios, C. et al., 2008. On the fragility of PI controllers for time-delay SISO systems. In *2008 16th mediterranean conference on control and automation*. pp. 529–534.
Mesbahi, A. & Haeri, M., 2014. Robust non-fragile fractional order PID controller for linear time invariant fractional delay systems. *Journal of Process Control*, 24(9), pp.1489–1494.
Michiels, W. et al., 2002. Continuous pole placement for delay equations. *Automatica*, 38(5), pp.747–761.
Michiels, W. & Vyhlidal, T., 2005. An eigenvalue based approach for the stabilization of linear time-delay systems of neutral type. *Automatica*, 41(6), pp.991–998.
Michiels, W., Vyhlidal, T. & Zitek, P., 2010. Control design for time-delay systems based on quasi-direct pole placement. *Journal of Process Control*, 20(3), pp.337–343.
Morarescu, I.-C. et al., 2011. Stability crossing boundaries and fragility characterization of PID controllers for SISO systems with I/O delays. In *Proceedings of the 2011 American Control Conference*. pp. 4988–4993.
O'Dwyer, A., 2009. *Handbook of PI and PID controller tuning rules*, World Scientific.
Onat, C., 2019. A new design method for PI-PD control of unstable processes with dead time. *ISA Transactions*, 84, pp.69–81.
Ou, L., Zhang, W. & Gu, D., 2006. Sets of stabilising PID controllers for second-order integrating processes with time delay. *IEE Proceedings-Control Theory and Applications*, 153(5), pp.607–614.
Pan, I. et al., 2011. Stabilizing gain selection of networked variable gain controller to maximize robustness using particle swarm optimization. In *Process Automation, Control and Computing (PACC), 2011 International Conference on*. pp. 1–6.
Pan, I. & Das, S., 2013. Frequency domain design of fractional order PID controller for AVR system using chaotic multi-objective optimization. *International Journal of Electrical Power & Energy Systems*, 51, pp.106–118.
Saha, S. et al., 2012. A conformal mapping based fractional order approach for sub-optimal tuning of PID controllers with guaranteed dominant pole placement. *Communications in Nonlinear Science and Numerical Simulation*, 17(9), pp.3628–3642.
Shafiei, Z. & Shenton, A., 1994. Tuning of PID-type controllers for stable and unstable systems with time delay. *Automatica*, 30(10), pp.1609–1615.
Silva, G.J., Datta, A. & Bhattacharyya, S.P., 2002. New results on the synthesis of PID controllers. *IEEE Transactions on Automatic Control*, 47(2), pp.241–252.
Silva, G.J., Datta, A. & Bhattacharyya, S.P., 2007. *PID controllers for time-delay systems*, Springer Science & Business Media.
Tan, N., 2005. Computation of stabilizing PI and PID controllers for processes with time delay. *ISA Transactions*, 44(2), pp.213–223.
Tang, Y. et al., 2012. Optimum design of fractional order PIλDμ controller for AVR system using chaotic ant swarm. *Expert Systems with Applications*, 39(8), pp.6887–6896.
Wang, D.-J., 2012. A PID controller set of guaranteeing stability and gain and phase margins for time-delay systems. *Journal of Process Control*, 22(7), pp.1298–1306.
Wang, Q.-G. et al., 2009. Guaranteed dominant pole placement with PID controllers. *Journal of Process Control*, 19(2), pp.349–352.
Wang, Q.-G., Liu, M. & Hang, C.C., 2007. Approximate pole placement with dominance for continuous delay systems by PID controllers. *The Canadian Journal of Chemical Engineering*, 85(4), pp.549–557.
Wang, Y.-G. & Shao, H.-H., 2000. PID auto-tuner based on sensitivity specification. *Chemical Engineering Research and Design*, 78(2), pp.312–316.

Zitek, P. & Fiser, J., 2018. A universal map of three-dominant-pole assignment for PID controller tuning. *International Journal of Control*, pp.1–8.
Zitek, P., Fiser, J. & Vyhlidal, T., 2013. Dimensional analysis approach to dominant three-pole placement in delayed PID control loops. *Journal of Process Control*, 23(8), pp.1063–1074.
Zitek, P., Fiser, J. & Vyhlidal, T., 2019. Dynamic similarity approach to control system design: delayed PID control loop. *International Journal of Control*, 92(2), pp.329–338.