Metaheuristic Tuning of Linear and Nonlinear PID Controllers to Nonlinear Mass Spring Damper System

Sudarshan K. Valluru and Madhusudan Singh

Department of Electrical Engineering, Delhi Technological University
Bawana Road, Delhi-110042, India.

Abstract

This paper makes an evaluation of the metaheuristic algorithms namely multi-objective genetic algorithm (MOGA) and adaptive particle swarm optimisation (APSO) algorithms to determine optimal gain parameters of linear and nonlinear PID controllers. The optimal gain parameters, such as $K_p$, $K_i$, and $K_d$ are applied for controlling of laboratory scaled benchmarked nonlinear mass-spring-damper (NMSD) system in order to gauge their efficacy. A series of experimental simulation results reveals that the performance of the APSO tuned nonlinear PID controller is better in the aspects of system overshoot, steady state error, settling time of the trajectory for the NMSD system.

Keywords: Linear and Nonlinear PID tuning; multi-objective genetic algorithm; adaptive particle swarm optimisation; mass spring damper system.

INTRODUCTION

PID controllers are extensively dominative and popular in closed loops of feedback dynamical systems due its simplicity and robustness. It is a very important ingredient and bread and butter of control of dynamical systems. PID controllers have witnessed many changes from analog controllers to digital controllers. Practically all PID controllers in the industry is microprocessor based digital PID controllers has given an opportunities to provide an additional degree of freedom corresponding to automatic tuning, gain scheduling, continuous adaption, and in recent times evolutionary meta-heuristic tuning. This steered researchers to explore the best method in finding optimum gain parameters by many tuning strategies are proposed for PID and PI controls for systems with monotonic transient responses. Usually, the PID controller dynamics can be employed by the linear control actions in the form of tracking error with fixed gains called as linear PID (L-PID)[1]. As an outcome, it often reduces the performance of bench scaled NMSD systems, if the controller operates over a wide envelop about the point of tuning and need to compromise among the overshoot and fastness of the responses. In order to avoid this ambiguity and to improve the performance of bench scaled NMSD system, nonlinear PID (NL-PID) was introduced with nonlinear characteristics[2][3]. This nonlinear combination can offer a further degree of freedom to attain an enhanced dynamical system performance. However, this enhancement can only be accomplished at the cost of higher complexity in the controller. Nature and bio-inspired metaheuristics can diminish some of the computational difficulties such as monotonous tuning via priori knowledge into the controller. Currently several metaheuristic algorithms such as genetic algorithms (GA) and particle swarm optimization (PSO) etc. have been proposed to estimate the optimised tuning gains of PID controllers[4] for linear systems. But these basic variants of GA and PSO algorithms have the tendency to premature convergence for multi-dimensional systems[5]. This has been the motivation in the direction to use improved variants of these algorithms, such as MOGA and APSO techniques to compute L-PID and NL-PID gains according to the bench scaled NMSD system, such that premature convergence and tedious tunings are diminished. In recent past several bio-inspired metaheuristics such as GA and PSO used for tuning of conventional PI and PID controllers [6]–[10]. Contrasting to the current literature, the work in this paper evaluates the MOGA and APSO tuning of L-PID and NL-PID controller for a multivariable system.

The paper organised in five sections commencing with an introduction followed section II, which analyses the modelling of mass spring damper system. Section III discuss the design of L- PID and NL-PID controllers. In section IV the methodology of MOGA and APSO tuning of L-PID and NL-PID controllers for NMSD system is presented. Section V presents the evolutionary analysis and results. Section VI provides the conclusion.

MODELLING OF NONLINEAR MASS SPRING DAMPER SYSTEM

A. Mathematical Modelling

The NMSD system is a fluctuating system mainly consisting of an element called the inertia or mass which stores energy in the form of kinetic energy, a damper, and a potential energy storing system i.e. stiffness with an energy dissipating element[11][12]. Due to the presence of these components in NMSD the nonlinearity is observed. In the vibratory system the contact abrasion among the various sliding parts can be insisted by damping effect or it may persist due to elastic distortion and internal friction[13]. The two above causes may not be wholly excluded as they are irrepressible. The damping coefficient ($\beta$) can be served as the final damping basis in the usage of mechanical viscid dampers to catch the necessary damping. The stiffening or weakening of the spring in the course of time, as it compresses and elongates in addition to the above stated causes leads to a nonlinear response[14][15] [16]in a practical system.

The bench scaled laboratory model shown in Figure1 (a)
actuated by an AC servomotor entails of an overall mass \((m)\) that glides steeply vacillating despite the fact being attached to a damper and a spring on its bottom side to a stiff plane. The free body diagram of the NMSD system with simple parallel connection of dashpot and spring in series with the mass shown in Figure 1 (b).

**Figure 1(a):** Photograph of the bench scaled laboratory nonlinear mass spring damper system energised with an AC servo motor (courtesy by dynamics lab Delhi Technological University).

**Figure 1(b):** Free body diagram of mass spring damper system

### B. Differential Equations

The general NMSD oscillatory system can be described by the equation (1)

\[
m \frac{d^2 x}{dt^2} + f \left( x, \frac{dx}{dt}, \beta, k \right) = 0
\]

The system mass or inertia is indicated by \(m\) (Kgf), damping constant through \(\beta\) (Kgf/cm/sec) and \(k\) symbolizes the spring constant in Kgf/cm. The function \(f\) is used to represent the nonlinear load displacement curve of the spring if some peripheral force \(F\) is operated upon the mass causing a certain displacement in the NMSD system then the equation (1) amends as controlled differential equation as (2)

\[
m \frac{d^2 x}{dt^2} + f \left( x, \frac{dx}{dt}, \beta, k \right) = F
\]

Since here nonlinear spring is considered and the spring stiffness function is given by

\[
k = k_1 x + k_2 x^3
\]

As a result the system nonlinear differential equation turn out to be

\[
m \frac{d^2 x}{dt^2} + k_1 x + k_2 x^3 + \beta \frac{dx}{dt} = F
\]

### C. State Space modelling

Bearing in mind the two state space variables for the system be \(x_1\) and \(x_2\) such that

\[
x_1 = x
\]

\[
x_2 = \frac{dx}{dt}
\]

Then

\[
x_1 = \frac{dx}{dt} = x_2
\]

\[
x_2 = \frac{d^2 x}{dt^2}
\]

By rearranging equation (4) to get equations (9) and (10)

\[
m x_2 + k_1 x_1 + k_2 x_1^3 + \beta x_2 = F
\]

\[
x_2 = - \frac{k_1 x_1}{m} - \frac{k_2 x_1^3}{m} - \frac{\beta x_2}{m} + \frac{F}{m}
\]

Then the system can be compactly written as

\[
x = f(x) + g(x)u + d
\]

Where

\[
f(x) = \left[ \begin{array}{c} x_2 \\ - \frac{k_1 x_1}{m} - \frac{k_2 x_1^3}{m} - \frac{\beta x_2}{m} \end{array} \right]
\]

\[
g(x) = \left[ \begin{array}{c} 0 \\ 1 \\ 0 \end{array} \right]
\]

\[d = 0\]

The bench scaled laboratory NMSD system was modelled in SIMULINK and is shown in Figure 2. The mass \(m\) is taken as 45.3 Kg, the spring stiffness coefficients are taken as \(k_1 = 10^{10}\) and \(k_2 = 3 \times 10^{11}\) and the damping coefficient \(\beta\) is considered as 17.83 kg/cm/sec.

**Figure 2:** Scheme of the Simulink modelling of NMSD system.
The displacement and velocity of the NMSD for a unity control signal are shown in Figure 2 (a) and 2(b). The trajectories of NMSD for various control signals ranging from 0.1 to 1 is plotted and shown in Figure 3. The open loop response of NMSD without any controller for 0.7 control signal is also shown in Figure 4.

Figure 2(a): NMSD system displacement with a control input of unity

Figure 2(b): NMSD velocity with a control input of unity

Figure 3: Trajectories of NMSD with different control signals

Figure 4: Open loop response of NMSD without controller for a control input of 0.7

DESIGN OF LINEAR AND NONLINEAR PID CONTROLLERS

Generally the PID controllers are used in feedback loops of industrial control systems to enhance their performance. The motivating factor to notice here that more than 95% of controllers deployed in control systems are either PID controllers[17][18] or their modified variants. The reason for the versatility is that once the mathematical model of the system can be obtained it would be easy to find the best gain values of the PID controllers to ensure the satisfactory transient and steady state responses. However, it is well known fact that 80% of the industrial PID controllers are not properly tuned according to system parameter variations due to the intricacy or tedious computation of PID gains, hence to overcome this impediment numerous schemes of PID tuning extending from simple Ziegler Nicholas tuning to various bio-inspired tuning techniques are under way.

A. Linear PID Controller

The L-PID controller relates the reference input to the real value of the system input, regulates the deviation and harvests a control signal to diminish the minimum deviation value. The spawned control signal \( u(t) \) is the linear difference between the reference input and the output of the system along with the combination of proportional, integral and differential coefficients. A general structural design of L-PID[19] is shown in Figure 5 which uses the reference signal to the mass spring damper system, the error between the actual output and the reference signal along with a proportional gain \( (K_p) \), integral gain \( (K_i) \) and differential gain \( (K_d) \) generates the control signal \( u(t) \) is given in equation (11)

\[
u(t) = k_p + k_i \int \! dt + k_d \frac{de}{dt}
\]  

(11)
signal for the system. This inspires the authors to propose an NL-PID [20] which behaves superior with system deviations and instabilities. Aimed at this, a general nonlinear PID module can be familiarized as in equation (13)

$$f(e, \alpha, \delta) = \begin{cases} |e|^{\alpha} \text{sign}(e) & |y| \geq \delta \\ \frac{e}{\delta^{\alpha}} & |y| < \delta \end{cases}$$

(13)

Where sign(e) = \begin{cases} 1, e \geq 0 \\ -1, e < 0 \end{cases}

\(e\) is the error signal, \(\delta\) defines the linear domain of the function \(f\), at this juncture function \(f\) can sustain a larger bound of nonlinear features can be made known by \(\alpha\). As a result now the control signal being spawned by the NL-PID is given in equation (14)

$$u(t) = K_p e f_p(e_p, \alpha_p, \delta_p) + K_i f_i(e_i, \alpha_i, \delta_i) \int e dt + K_d f_d(e_d, \alpha_d, \delta_d) \frac{de}{dt}$$

(14)

The function \(f(\alpha, \delta, e)\) represents the error feedback rate to recompense the system non linearity considering the \(\alpha_p\) value chosen in the domain of \(\alpha_p \epsilon [0,1]\) since the requirement is poorer gain once the error jumps high and vice versa. The integral saturation difficulty of the integral term can be resolved by means of \(\alpha_i\) value in the domain of \(\alpha_i \epsilon [-1,0]\). The value of the derivative term is taken as \(\alpha_d > 1\) so that when the steady state is touched, the outcome of the derivative term is lessened. The systematic structure design of NL-PID is shown in Figure 6.

\[\begin{align*}
 J(x) & = \int [e(x, t)]^2 dt \\
 & \text{Where } x = [k_p, k_i, k_d]
\end{align*}\]

\(\text{Equation (12)}\)

### B. Non-Linear PID Controller

Mostly the PID controllers are the best-employed controllers for control of NMSD system. While L-PID controllers are often sufficient for controlling NMSD system during small linear operating envelop but for high-performance control, the competency and robustness of L-PID control are insufficient. The key disappointment of L-PID controller arises due to the limitations in the mathematical model i.e. conflict amid augmented gain and the faster response of the system. But if a appropriate regulation can be articulated these boundaries would be eliminated, thus paving the way to a desired control

**METAHEURISTIC TUNED L-PID AND NL-PID CONTROLLERS**

Metaheuristic algorithms are different from the traditional optimisation algorithm by the fact that the later uses only a solitary point result while a metaheuristic algorithm uses a significant populace for obtaining an optimal solution. Metaheuristic algorithms can all together assess a large number of points in the hunt space concurrently as they are intrinsically parallel in process which qualifies them to regard as numerous points in the hunt space at the same time not consuming abundant time to converge. Also, they are more liable to converge to global optima relatively than local optima. For the duration of an iteration of metaheuristic algorithm, there is an economical collection of the better-resulting individuals from the hunt space which springs enhanced result and the remaining
solutions are thrown away. Further, some metaheuristic algorithms also spawn new points or offspring to enhance the hunt space as well as the solution. Off late numerous metaheuristic algorithms have been advanced like Genetic algorithm, Firefly algorithm, Ant colony algorithm and particle swarm optimizer algorithm with different variants etc. All these algorithms contrast from one another on the basis of offspring generation, variables to be tuned and replacement mechanism. For multivariable non-linear dynamical systems, standard GA and PSO are facing premature converges[21]. So, in order to diminish this problem the modified variants such as MOGA and APSO have been suggested for tuning of PID controllers, since when using the metaheuristic algorithms the limitation to the Eigen values of the nonlinear system matrix does not apply.

C. Multi-Objective Genetic Algorithm
The Multi-Objective Genetic algorithms were one type of combinatory optimised search algorithm which requires discretized search space in order to solve real decision variables of system optimisation problems for both single objectives as well as multi-objective functions[22]. The MOGA selects individuals based on random selection from the population and consists of genetic transformations such as mutation, crossover to create a new set of population. Hence initially there is no clue, regarding the best answer for an optimisation problem. Over the continuous generations, the finest among the lot are selected, operators are functioned upon them and appraised till reaching the global or suboptimal solutions. The theme of using MOGA here is to obtain the optimal value of L-PID and NL-PID parameters (K_p, K_i and K_d) in a way the objective functions J(x) are minimised. The global outcome of the findings are optimal gains for the tuning of the PID controller and concurrently enhance in the transient as well as the steady-state response of the NMSD system. The objective function J(x) befits for PID controllers with best gain parameters, is given in equation (15)

\[ J(x) = (r(n) - y(n))^2 + (max[r] - max[y])^2 \]  

Where r(n) is the n^{th} input of the system and y(n) is the n^{th} output of the system. A general block diagram of MOGA tuned L-PID or NL-PID is shown in Figure7 which uses the torque as a reference signal to the NMSD system, the error between actual output and the reference signal along with proportional(K_p), integral(K_i), derivative (K_d) gains generate the control signal. The control signals of MOGA tuned L-PID and NL-PID controllers for NMSD is shown in Figure 8 and Figure 10 respectively.

![Figure 7](image-url)  

Figure 7: MOGA tuned L-PID/NL-PID controller for NMSD system

The pseudo code of MOGA employing to tune L-PID and NL-PID controllers for NMSD system can be summarised as follows:

1. Fix the range of PID controller parameters as objective function \( f(x) = (K_p, K_i, K_d)^T \).
2. Encode the solution into chromosomes as binary strings.
3. Define the fitness \( F \propto f(x) \) for maximisation.
4. Generate the initial population of size N for the PID controller parameters.
5. Initial probabilities of crossover (p_c) and mutation (p_m).
6. Evaluate the initial population.
7. While \( t < \text{Max. number of generations} \)
   Generate new solution by crossover and mutation
   If \( p_c > \text{rand} \), Crossover; \text{end if}
   If \( p_m > \text{rand} \), Mutation; \text{end if}
   The new solutions are accepted if fitness increases.
   Select the current best for new generation
   \text{end while.}
8. Decode the binary strings and visualise the new values of \( K_p, K_i, K_d \).

The size of the population, dimension, mutation rate, selection rate, maximum iterations used in the MOGA process to find best optimised objective parameters of L-PID and NL-PID controllers for NMSD system are taken as 10,3,0.2,0.3,100 respectively. The estimated L-PID and NL-PID controller gain parameters were optimised by multi-objective genetic algorithm, which is tabulated in Table.1. The MOGA tuning response trajectories of the NMSD system with L-PID and NL-PID is shown in Figure9 and Figure11 respectively.

<table>
<thead>
<tr>
<th>Controller</th>
<th>( K_p )</th>
<th>( K_i )</th>
<th>( K_d )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear-PID</td>
<td>43.147155</td>
<td>10.772516</td>
<td>7.670948</td>
</tr>
</tbody>
</table>

TABLE 1: MOGA TUNED L-PID AND NL-PID CONTROLLER BEST GAIN PARAMETERS
Adaptive Particle Swarm Optimization Algorithm

APSO algorithm is an effective algorithm for optimising a wide range of objective functions, inspired from the swarming phenomena of fish schooling and bird flocking[23]. The multiple agents known as particles, swarm around the search space starting from an initial guess. APSO has some comparisons with MOGA, but APSO is much simpler because it uses real number of randomness and global communications instead of mutation/crossover[24]. The APSO algorithm starts with the initialization of group of arbitrary particles in the hunt space, and individual particle signifies a potential solution for the optimisation of an objective function.

For the duration of each iteration, all the particles in the hunt space ascertain a probable solution. Later this, the particle apprises its position conferring to the velocity vector which comprises its previous velocity and is decided by taking in interpretation the past local and global best solutions. Then the best solutions are retained and the particle heads towards local best solution accomplished by its peer particles but also the global best. Hence if a particle has learned a new best solution then all other particles would try to leap toward it. Here the four important terms in APSO for the particles in the swarm are $p_i$ (position), $v_i$ (current velocity), $P_{b_i}$ (local best position) and $G_{b_j}$ (global best position). Individual particle is reorganized according to the above four features in each iteration, supposing the cost function $J$ to be minimised. The innovative velocity of a particle is determined by the equation (16)

$$v_{i,j}(n + 1) = C \times \{wv_{i,j}(n) + c_1r_1[P_{b_{ij}}(n) - p_{ij}(n)] + c_2r_2[G_{b_{j}}(n) - p_{ij}(n)]\}$$  (16)

Where $n$ is the number of iterations, $w$ is inertia weight, $c_1$ and $c_2$ are the acceleration coefficients (called cognitive and social component respectively) and $r_1$ and $r_2$ are two quasi-uniform random numbers between (0, 1). $v_{i,j}$ is the velocity of the $j^{th}$ dimension of the $i^{th}$ particle. The new position of the particle is updated by using equation (17)

$$p_{ij}(n + 1) = p_{ij}(n) + v_{ij}(n + 1)$$  (17)

The local best of every particle is updated by the equation (18)

$$P_{b}(n + 1) = \begin{cases} P_b(n), & J(p(n + 1)) \geq J(P_{b}(n)) \\ P_b(n + 1), & \text{otherwise} \end{cases}$$  (18)

Here the population size is represented by $s$, then the optimised global best can be obtained by the equation (19)
\[ G_b(n + 1) = \min_{P_{b_i}} J(P_b(n + 1)), \quad 1 \leq i \leq s \] (19)

Tuning of L-PID and NL-PID can be efficiently done by using particle swarm optimizer to find the values of optimal gain parameters \((K_p, K_i, K_d)\) which will minimise the objective function \(J\). The block diagram of APSO tuned L-PID or NL-PID for NMSD system is shown in Figure 12 which uses the step reference signal to the NMSD system, the error between actual output and the reference signal along with proportional \((K_p)\), integral \((K_i)\), derivative \((K_d)\) gains generated the control signal \(u(t)\). The control signals of APSO tuned L-PID and NL-PID controllers for NMSD is shown in Figure 13 and Figure 15 respectively.

![Figure 12: APSO tuned L-PID/NL-PID controller for NMSD system](image1)

The APSO pseudo code to tune the L-PID and NL-PID controllers for NMSD system is summarised as follows:

1. Fix the range of PID controller parameters \(J(x), x = (K_p, K_i, K_d)^T\)
2. Initialize the locations of \(p_i\) and velocity \(v_i\) for swarm size \(n\), acceleration factors \(c_1\) and \(c_2\) to the controller parameters \(K_p, K_i\) and \(K_d\) each representing dimensions of particle position vector in real parameter space.
3. for each particle, \(i\) do each particle is tested for fitness by evaluating the objective function \(J(x)\).
4. Set the initial local best \((Pb_i)\), and global best \((Gb)\)
5. While stopping criteria is not reached do for every particle \(i\) do Generate innovative velocity \(v_{i,j}(n + 1)\) according to equation (36)
   Generate new positions \(p_{i,j}(n + 1)\) according to equation (37) Evaluate objective functions at new positions Renew the current best of each particle end for
   Find the current global best end while
6. Output the final results of optimised global best

The size of the swarm, dimension, maximum iterations, cognitive factor, social factor, constriction factor used in the adaptive particle swarm optimizer to find the best optimised objective parameters for L-PID and NL-PID controllers for the NMSD system are 10, 3, 50, 2.05, 2.05, and 2 respectively. The best gain parameters of APSO tuned L-PID and NL-PID controllers were given in Table II. The APSO tuning response trajectories of the NMSD system with L-PID and NL-PID is shown in Figure 14 and Figure 16 respectively.

<table>
<thead>
<tr>
<th>Controller</th>
<th>(K_p)</th>
<th>(K_i)</th>
<th>(K_d)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear-PID</td>
<td>233.8365</td>
<td>43.0993</td>
<td>1.2873</td>
</tr>
<tr>
<td>Nonlinear-PID</td>
<td>3.7243</td>
<td>0.4074</td>
<td>12.6238</td>
</tr>
</tbody>
</table>

![Figure 13: Control signal from L-PID controller with APSO tuning](image2)

![Figure 14: Response of APSO tuned L-PID for NMSD system](image3)

![Figure 15: Control signal from NL-PID controller with APSO tuning](image4)
EVOLUTIONARY ANALYSIS AND RESULT

The simulation study of benchmarked NMSD trajectory response using MOGA, APSO tuned L-PID and NL-PID controllers are presented. The step reference trajectory considered as model reference for NMSD system together with transient behaviour has been taken account. The system is originally fluctuating system with unmodelled dynamics and parameter variations. Figure4, Figure9, Figure11, Figure14 and Figure16 shows the system trajectory tracking response without controller, MOGA tuned L-PID and NL-PID controllers and also APSO tuned L-PID and NL-PID controllers in the aspects of various performance criteria like system overshoot, steady state error and settling time are summarised in Table. III.

TABLE III. CONTROLLER PERFORMANCE PARAMETERS OF NMSD

<table>
<thead>
<tr>
<th>Type of Controller</th>
<th>Percentage overshoot</th>
<th>Steady state error</th>
<th>Settling time (Sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absence of controller</td>
<td>34.9192</td>
<td>0.0050</td>
<td>0.0389</td>
</tr>
<tr>
<td>MOGA tuned L-PID</td>
<td>0.7628</td>
<td>0.0021</td>
<td>0.4012</td>
</tr>
<tr>
<td>MOGA tuned NL-PID</td>
<td>0.4736</td>
<td>0.0012</td>
<td>0.3494</td>
</tr>
<tr>
<td>APSO tuned L-PID</td>
<td>--</td>
<td>0.0003</td>
<td>0.2043</td>
</tr>
<tr>
<td>APSO tuned NL-PID</td>
<td>--</td>
<td>0.0003</td>
<td>0.0128</td>
</tr>
</tbody>
</table>

CONCLUSION

The MOGA, APSO tuned linear and nonlinear PID controllers have been successfully developed and tested for their efficacy in terms of steady state error, overshoot and settling time on a nonlinear mass spring damper system. The efficacy of metaheuristics such as MOGA and APSO tuned NL-PID controller are better in comparison to the L-PID controlled NMSD system. It proves that the NL-PID controller mends the performance of the NMSD system with a reduction in both the overshoot as well as the settling time as related to the MOGA, APSO tuned L-PID controller’s responses. Thus ultimately APSO tuned NL-PID controller gives the finest outcome.

REFERENCES


