Design of Self-Tuning PID Control in a Mechanisms System

1 Rong-Fong Fung and 2 Chih-Cheng Kao
1 Department of Mechanical and Automation Engineering National Kaohsiung First University of Science and Technology
2 Department of Electrical Engineering Kao Yuan Institute of Technology

Abstract: In this paper, a novel design method for self-tuning PID controller of in mechanisms system using the particle swarm optimization (PSO) algorithm is presented. This paper demonstrated in detail how to employ the PSO to search efficiently the optimal PID controller parameters of in mechanisms system. The proposed approach had superior features, including easy implementation, stable convergence characteristic, and good computational efficiency. Fast tuning of optimum PID controller parameters yields high-quality solution. Using the PSO approach, the initial PID parameters under normal operating condition can be found out. By the same way, the best parameters of PID controller under full-load condition can be found, too. The proposed self-tuning PID controller will automatically tune its parameters under these ranges. Moreover, the PC-based controller is implemented to control the position of the motor mechanism coupling system. The simulation and experimental results will show the potential of the proposed controller.

Indexing terms: mechanisms system, self-tuning PID controller, particle swarm optimization, optimal control.

I. Introduction

During the past decades, the process control techniques in the industry have made great advances. Numerous control methods such as adaptive control, neural control, and fuzzy control have been studied [1-5]. Among them, the best known is the proportional integral derivative (PID) controller, which has been widely used in the industry because of its simple structure and robust performance in a wide range of operating conditions. Unfortunately, it has been quite difficult to tune properly the gains of PID controllers because many industrial plants are often burdened with problems such as high order, time delays, and nonlinearities [1-6]. Over the years, several heuristic methods have been proposed for the tuning of PID controllers. The first method used the classical tuning rules proposed by Ziegler and Nichols. In general, it is often hard to determine optimal or near optimal PID parameters with the Ziegler-Nichols formula in many industrial plants [1-3].

For these reasons, it is highly desirable to increase the capabilities of PID controllers by adding new features. Many artificial intelligence (AI) techniques have been employed to improve the controller performances for a wide range of plants while retaining their basic characteristics. AI techniques such as neural network, fuzzy system, and neural-fuzzy logic have been widely applied to proper tuning of PID controller parameters [1-2].

Particle swarm optimization (PSO), first introduced by Kennedy and Eberhart, is one of the modern heuristic algorithms. The PSO technique can generate a high-quality solution within shorter calculation time and stable convergence characteristic than other stochastic methods [7-9]. Much research is still in progress for proving the potential of the PSO in solving complex power system operation problems. Because the PSO method is an excellent optimization methodology and a promising approach for solving the self-tuning PID controller parameters problem. This controller is called the PSO self-tuning PID controller.

The PID method is the most popular controller up to now. Despite the progression of many control theories, the PID controller is still the majority of industrial processes [11-13]. Due to the easily understanding of the physical sense for parameters of PID controller, engineers used to apply it to practical objects. However, the PID controller is not robust to wide parameter varying and large external disturbance. Especially for the highly coupling nonlinear system, the PID controller is lack of adaptive capability. Usually, the parameters of PID controller are manually tuned under ideal condition, that is the operating point without load. However, these parameters are mostly not suitable for the condition with full load. To achieve practical requirement, engineers have to adjust the parameters under different operating conditions. However, the robustness is limited with a small range. A rule to overcome this disadvantage is called self-tuning rule. Many researches and reports of self-tuning PID have been published. The parameter tuning at any time instance is usually based on a structurally fixed mathematical model produced by on-line identification procedure [14-15]. Unfortunately, recent plants are mostly difficult to obtained their fixed mathematical models. This paper proposed an intelligent self-tuning PID controller. Engineers will easily accept the straightforward design procedure. At the same time, the robustness will be expended to large range.

In this paper, a practical high-order mechanisms system with a PID controller is adopted to test the performance of the proposed PSO self-tuning PID controller. This paper proposes a self-tuning PID control method to the position control of slider-crank mechanism.
II. Slider Crank Actuated by a PM Synchronous

2.1 PM Synchronous

A model of a PM synchronous motor can be simplified to the following block diagram.

![Block Diagram of a PM Synchronous Motor](image)

Fig 1 Block Diagram of a PM synchronous motor

Usually, the PM synchronous motor is coupled with a gear speed reducer with a gear ratio of \( n \). Hence, the applied torque can be described as

\[
T = nK_i \omega_n - nJ_m \omega_n + nB_m \dot{\omega}_n
\]

where \( T \) is the torque applying in the direction of \( \omega_n \), \( K_i \) is the torque constant, \( J_m \) and \( B_m \) are the inertia and viscous damping ration, respectively.

2.2 Slider Crank Mechanism

In this section, Hamilton's principle and Lagrange multiplier are used to derive the differential equation for the slider-crank mechanism. The slider crank mechanism system is shown in Fig. 2 [16].

![Slider Crank Mechanism System](image)

Fig 2 A Slider Crank Mechanism System

The slider-crank mechanism consists of three parts: crank, rod and slider. This section establishes dynamic model of the slider-crank mechanisms system using Largange method. Through reduction and incorporation, we can derive the dynamic equation is [16]:

\[
M \dot{\omega}_n + N \omega_n + BU = D + \tau
\]

where

\[
M = \frac{1}{2}m_1 \omega_n^2 + m_2 \omega_n \sin^2 \theta + n^2J_m + \frac{1}{2}m_3 \omega_n \tau \sin \theta
\]

\[
N = \frac{1}{2}m_1 \omega_n \tau \sin \theta + \frac{1}{2}m_2 \omega_n \tau \sin^2 \theta + \frac{1}{2}m_3 \omega_n \tau \sin^2 \theta
\]

\[
B = nK_t
\]

\[
U = i_q
\]

D

\[
F_{BE} r \sin \theta
\]

\[
F_{BE} l \sin \theta
\]

where \( m_1 \), \( m_2 \), and \( m_3 \) are the mass of crank, rod and slider, respectively, and \( r \) and \( l \) are length of crank and rod, and \( \theta \) and \( \phi \) are angle of crank and rod.

The translation position \( X_B \) can be obtained by transforming \( \theta \), that is

\[
X_B = r \cos \theta \cos \phi \cos \gamma - r \cos \theta \sin \phi \cos \lambda
\]

\[
l \sin \phi \sin \gamma
\]

III. Particle Swarm Optimization

In 1995, Kennedy and Eberhart first introduced the particle swarm optimization (PSO) method. The method has been found to be robust in complex system, which is derived from the social-psychological theory. Instead of using evolutionary operators to manipulate the particle (individual). Each particle is treated as a volume less than \( g \)-dimensional search space and keeps track of its coordinates in the problem space, which are associated with the best solution (evaluating value) it has achieved so far [7-10]. This value is called pbest. Another best value that is tracked by the global version of the particle swarm optimizer is the overall best value, and its location, obtained so far by any particle in the group, is called gbest.

The PSO concept consists of, at each time step, changing the velocity of each particle toward its Pbest and gbest locations. For example [9-10], the jth particle is represented as \( x_j = (x_{j,1}, x_{j,2}, \ldots, x_{j,g}) \) in the g-dimensional space. The best previous position of the jth particle is recorder and represented as pbest\(_{j,g} = (\text{pbest}_{j,1}, \text{pbest}_{j,2}, \ldots, \text{pbest}_{j,g}) \). The index of best particle among all of the particles in the group is represented by the gbest\(_g\). The rate of the position change (velocity) for particle \( j \) is represented as \( v_j = (v_{j,1}, v_{j,2}, \ldots, v_{j,g}) \). The modified velocity and position of each particle can be calculated using the current velocity and distance from pbest\(_{j,g} \) to gbest\(_g \) as shown in the following formulas:

\[
v^{(t+1)}_{j,g} = v^{(t)}_{j,g} + c_1 \text{rand}() \cdot (\text{pbest}_{j,g} - x^{(t)}_{j,g})
\]

\[
v^{(t+1)}_{j,g} = v^{(t)}_{j,g} + c_2 \text{rand}() \cdot (\text{gbest}_{g} - x^{(t)}_{j,g})
\]

where

\[
\begin{align*}
&n \quad \text{number of particles in a group;} \\
m \quad \text{number of members in a particle;} \\
t \quad \text{pointer of iterations (generations);} \\
v^{(t)}_{j,g} \quad \text{velocity of the particle } j \text{ at iteration } t, \\
V^{\text{min}}_g \quad \text{minimum velocity of particle } g, \\
V^{\text{max}}_g \quad \text{maximum velocity of particle } g, \\
w \quad \text{inertia weight factor;} \\
c_1, c_2 \quad \text{acceleration constant;} \\
\text{rand}(), \text{Rand}() \quad \text{random number between 0 and 1;}
\end{align*}
\]
In the above procedures, the parameter $V^\text{max}$ determined the resolution, or fitness, with which regions be searched between the present position and the target position. If $V^\text{max}$ is too high, particles might fly past good solution. If $V^\text{max}$ is too small, particles may not explore sufficiently beyond local solution.

The PSO algorithm was mainly utilized to determine three optimal controller parameters $kp$, $ki$, and $kd$. We defined three controller parameters to compose an individual $K$ by $K := [kp, ki, kd]$; its dimension is $n \times 3$. The searching procedures were shown as below:[10]

Step 1) Specify the lower and upper bounds of the three controller parameters and initialize randomly the individuals of the population including searching points, velocities, pbests, and gbest.

Step 2) Calculate the evaluation value of each individual in the population.

Step 3) Compare each individual’s evaluation value with its pbest. The best evaluation value among the pbest is denoted as gbest.

Step 4) Modify the member velocity $v$ of each individual $K$ according to

\[
v^{(t+1)}_{j,g} = v^{(t)}_{j,g} + c^*_v\text{rand}(\cdot)(pbest_{j,g} - k^{(t)}_{j,g})
\]

Step 5) If $v^{(t)}_{j,g} < V^\text{min}_{j,g}$, then $v^{(t+1)}_{j,g} = V^\text{min}_{j,g}$.

Step 6) If $v^{(t)}_{j,g} > V^\text{max}_{j,g}$, then $v^{(t+1)}_{j,g} = V^\text{max}_{j,g}$.

Step 7) If the number of iterations reaches the maximum, then go to Step 8. Otherwise, go to Step 2.

Step 8) The individual that generates the latest gbest is an optimal controller parameter.

IV. Fuzzy Self-Tuning PID Controller

The structure of the proposed fuzzy PID controller is shown in figure 3. The detailed design procedures are described in the following sections.
By the same conception, the integral controller is helpful for steady state and hurtful for transient state. Hence, the integral (I) controller’s gain should be increased along with the error and error derivation decreasing. The curve is shown as figure 5. Let the tuning rule is defined as

\[ K_I t \quad K_I^0 \quad K_I^1 \quad E(t) , \dot{E}(t) \]  

(7)

Let \( K_I^0 \), \( I_{\min} \), \( K_I^1 \), \( I_{\max} \) and \( I_{\max} \) and \( I_{\min} \) are the gains under no-load and full-load conditions respectively.

Large differential controller can increase the speed of response. At the same time, large differential controller will cause large steady state error. Therefore, the differential (D) controller’s gain should be decreased along with the error and error derivation decreasing. The tuning rule can be shown as following figure and equation.

\[ K_D t \quad K_D^0 \quad K_D^1 \quad E(t) , \dot{E}(t) \]  

(8)

Let \( K_D^0 \), \( D_{\min} \), \( K_D^1 \), \( D_{\max} \) and \( D_{\max} \) and \( D_{\min} \) are the gains under full-load and no-load conditions respectively.

4.2 Adjusting Factor

In order to intelligently and automatically tune the controller’s parameters, the adjusting factor is obtained by fuzzy theory. By fuzzy control theory, the first work is to defuzzify the reference variables. The reference variables in designing procedure are position error and position error difference. There are seven linguistic variables used in this paper for each reference variables, that are large negative (LN), medium negative (MN), small negative (SN), zero (ZO), small positive (SP), medium positive (MP) and large positive (LP). The membership functions representing these linguistic expressions are defined as triangular type function shown in figure 7. In order to process the reference variable more efficiently, the position error is deal with linear convert from \(+0.05m~+0.05m\) into \(0~1\). At the same time, the position error difference is converted from \(+100m/s ~+100m/s\) to \(0~1\). The output factor is restricted in \(0~1\), shown in figure 7, too.

Secondly, the fuzzy relation and fuzzy rule are defined as 49 if-then rules for fuzzy inference. Due to the common sense, if the absolutions of error and error difference are large, then the adjusting factor needs to be large. Respectively, if the absolutions of error and error difference are small, then the adjusting factor is small, too.
The if-then rules can be established as followings.

Rule1: If \( e \) is LN and \( \dot{e} \) is LN, then \( t \) is LP
Rule2: If \( e \) is MN and \( \dot{e} \) is MN, then \( t \) is MP
Rule3: If \( e \) is SN and \( \dot{e} \) is SN, then \( t \) is SP

The if-then rules can be tabulated as follow table.

<table>
<thead>
<tr>
<th>( \dot{e} )</th>
<th>( e )</th>
<th>t</th>
</tr>
</thead>
<tbody>
<tr>
<td>LN</td>
<td>LN</td>
<td>LP</td>
</tr>
<tr>
<td>MN</td>
<td>MN</td>
<td>MP</td>
</tr>
<tr>
<td>SN</td>
<td>SN</td>
<td>MP</td>
</tr>
<tr>
<td>ZO</td>
<td>ZO</td>
<td>SP</td>
</tr>
<tr>
<td>SP</td>
<td>SP</td>
<td>LP</td>
</tr>
<tr>
<td>MP</td>
<td>MP</td>
<td>MP</td>
</tr>
<tr>
<td>LP</td>
<td>LP</td>
<td>LP</td>
</tr>
</tbody>
</table>

Finally, by the centroid defuzzification method, the adjusting factor is calculated from

\[
\alpha_{i} = \frac{1}{\sum_{i}^{n} \mu_{i}(t,e)} \int_{0}^{t} \int_{e} \mu_{i}(t,e) \, dt \, de
\]

where \( \alpha \) means the each central value of the linguistic variables of adjusting factor. Finally, applying the adjusting factor to eq (6)~(8), it will form the intelligent self-tuning PID controller.

V. Simulation Results

In this section, numerical simulation results are used to demonstrate the potential of the proposed control rule. To demonstrate the performance, the PID self-tuning method is compared with a fixed PID control. The actual slider crank mechanism dimensions are \( m_{1}=0.05454, \, m_{2}=0.2795, \, m_{3}=0.16, \, r=0.056, \, R=0.086, \, l=0.174, \, l'=0.055, \, K_{r}=0.6732, \, J_{m}=0.00062 \) and \( B_{m}=0.000153 \). The objective is to control the desired periodically translation position from 0.2107m to 0.1858m. The desired specifications are settling time \( t_{s}=0.5 \) sec, rising time \( t_{r}=0.25 \) sec, maximum overshoot \( M_{p}<5\% \) and steady state error \( e_{ss}<1\% \).

Figures 8(a) show the response of fixed PID. Using the PSO approach, we can get the parameters for optimal performance under no-load condition, where \( K_{P}^{no}=3.5695, \, K_{I}^{no}=2.9267, \, K_{D}^{no}=0.7727 \). The parameter variation is appeared by the mass of slider crank \( m_{3} \) changed from 0.16Kg to 0.8Kg. The external force is changed from 0Nt to 5Nt. While the load and parameter variation not existed, the performance can satisfy the requirement. However, compared with these curves the response will not match the desired specification when the load or parameter varying existed. Obviously, the fixed PID obtained under no-load condition cannot achieve the robustness with parameter varying and load existed.

By the same way, the PID’s parameters for full-load condition, the optimal parameters for PID controller with full-load are \( K_{P}^{full}=4.1942, \, K_{I}^{full}=1.2997, \, K_{D}^{full}=1.1313 \). Theoretically, the dynamic response under full-load will match the requirements. It is also proven in figure 9(a). However, once the parameter varying and external load are removed, the responses will cause the large steady-state error and unexpected transient state shown in figure 9. The large steady-state error and overshoot appeared to show the bad robustness of fixed PID.

The proposed self-tuning PID controller is based on these two optimal parameters. Let the normal parameters are based on no-load condition, that is \( K_{P}^{no} = 3.5695, \, K_{I}^{no} = 2.9267, \, K_{D}^{no} = 0.7727 \). The tuning ranges are selected as \( K_{P}^{full} = 0.6247, \, K_{I}^{full} = 1.6270, \, K_{D}^{full} = 0.3586 \). According to the position error and position error difference, the fuzzy rule will automatically find the optimal adjusting factor. Applying eqs (6)~(8), the responses of the proposed control rule are shown in figure 10(a). Obviously, the proposed self-tuning PID controller has great robustness. Under different operating conditions, it can still maintain the desired performance.
Fig 8 Response of Translation Position with Fixed PID Control (Parameters Obtained Under No-Load Condition)

Fig 9 Response of Translation Position with Fixed PID Control (Parameters Obtained Under Full-Load Condition)

© Experimental Result with 0.8 kg External Load

(a) Simulation results

(b) Experimental Result with No-Load

(b) Experimental Result with 0.8 kg External Load

© Experimental Result with No-Load

(a) Simulation Results
VI. Experimental Results

In order to demonstrate the proposed control rule, a PC-based experimental equipment is setup in this paper. The experimental instrument of slider crank is divided into three parts: actuator, slider crank and controller. The photographic is shown in figure 11. The first part consists of a PM Synchronous motor, driver. The driver is worked on 3-phase, 220 V and 60 Hz. The slider crank is coupled with the PM motor. The translation position is measured by a photometer. The output of photometer scalar is 20000 pulse/m, which mapped to real translation position is 0~0.11m. To carry out the parameter varying and external load is to add an external mass (0.8Kg) on slider. The controller is based of a PC with Pentium-586 CPU. The PC plays the role of software development and data process. The data acquisition interface card (Advantech CO., PCL-1800) is installed in the ISA bus to handle the A/D and D/A process. The graphical software of Simulink is used to implement the proposed control rule. At the same time, the linear converts between the physical scale and voltage from sensors are also worked in this software.

VII. Conclusion

This paper proposed a simple scheme of fuzzy PID method. Based on the PSO approach, the nominal values and tuning ranges of PID parameters can be accuracy get. Applying the intelligent fuzzy rule, the adjusting factor can be tuned on-line. A PC-based controller is implemented to apply to the translation position control of a slider crank mechanism. Simulation and experimental results show the proposed controller is more robust to against parameter variation and external load.

Reference


[3] disturbance rejection PID controllers using genetic


