Controller Optimization using Cuckoo Search Algorithm of a Flexible Single-Link Manipulator

Ali Abdulhussain Al-Khafaji
Department of Applied Mechanics & Design,
Fakulti Kejuruteraan Mekanikal
Universiti Teknologi Malaysia
8130, Johor Bahru
MALAYSIA
ali1977abdulhussain@gmail.com

Intan Z. Mat Darus
Department of Applied Mechanics & Design,
Fakulti Kejuruteraan Mekanikal
Universiti Teknologi Malaysia
8130, Johor Bahru
MALAYSIA
intan@fkm.utm.my

Abstract- This study presents an optimization algorithm to find the optimal value of PD controller parameters of the well-known joint-based collocated (JBC) control scheme which used to control the flexible manipulator. The single link flexible manipulator is modeled using Finite element method (FDM) and Lagrangian approach. An evolutionary Cuckoo Search Algorithm was utilized to optimize the controllers’ parameters. Behavior of controlled system response including hub angle and end-point displacement are recorded and assessed. It was noted that the proposed Cuckoo Search Algorithm algorithm is effective to find the optimal value of the PID based controller which was able to achieve better optimum value, and result in better time-domain hub-angle response.

Key-Words: - Cuckoo Search algorithm, Flexible single-link Manipulator, finite element method

I. INTRODUCTION

Due to the extraordinary request for automation and safety, robotic manipulators are currently used in many applications to accomplish difficult jobs at dangerous places. Maximum stiffness rigid robotic manipulators leads to large robot weight to payload ratio, increased energy consumption, increased size of actuator, low speed and increased overall cost. To cover these problems, the design of flexible links manipulators with lightweight has been motivated. The flexible link manipulator system, comparing with rigid link manipulator system offers significant advantages for their rigid counterparts, include low inertia, light weight, few powerful actuators, cheaper construction, fast in response, safer operation, higher payload carrying capacity, more compact design and longer reach. Moreover, they can quickly be adapted to changing situations and product design variations. For such advantages, modern flexible link manipulator system are widely used in many areas such as the space vehicles, medical, defense and automation industries. The flexible manipulators vibration, due to structural flexibility is a major ongoing and an unwanted feature. Thus, to achieve the benefits of flexible manipulators, accurate models and efficient control methods have to be established [1][2][3].

The control objectives in a flexible single-link manipulator system are to control the angular displacement of the hub and to suppress vibration at the end point. A proportional-derivative (PD)-like controller called joint-based collocated (JBC) control (JBC-PD) has been used by several researchers to control the angular displacement of the hub of a single-link flexible manipulator [4],[5],[6],[7]. In JBC-PD, there are two parameters need to tune that is the proportional gain in the feed forward path of hub-angle reference input and derivative gain in hub-velocity feedback path. The researcher mentioned above used the root locus approach to design the controller, the two parameters of JBC-PD are obtained by finding the poles of the closed loop transfer function of the plant graphically [8]. Alternative method which can be used to find optimal parameters of JBC-PD controller is to utilize an optimization algorithm. H. Supriyono used an optimization algorithm referred to as Hybrid spiral dynamics bacterial foraging (HSDBF) to find the optimal parameters of JBC-PD controller for a flexible manipulator system[3].

Recently, a new metaheuristic algorithm, called Cuckoo Search (CS), has been developed by Yang and Deb (2009). CS has been shown to yield promising outcomes for solving various engineering optimization problem. CS is based on the interesting breeding behavior such as brood parasitism of certain species of Cuckoos, and the preliminary studies show that it is very promising and could outperform the existing algorithms such as Genetic Algorithm (GA), Particle swarm optimization (PSO) [9],[10],[11].

S.Sanajaoba Singh and Nidul Sinha presents an integral plus double derivative (IDD) in coordination with thyristor controlled phase shifter is applied in automatic generation control system. In this paper, CS is used to optimize the controllers gains in compare with Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). The comparison of convergence characteristics of GA, PSO and CS algorithms reveal faster convergence in case of CS algorithm, which is an obvious choice of reducing computational burden[12]. In this paper CS is implemented for adjusting the parameters of JBC-PD control which used for controlling the hub-angular displacement. CS has been adopted because it has several advantages, for instance it is able to find optimal parameters as well as to avoid being trapped in local minima.

In this research, CS is implemented for adjusting the parameters of JBC-PD control which used for controlling the hub-angular displacement. CS was chosen because it has advantage as it is able to find optimal parameters as well as avoid being trapped in local minima. The objective of this study is to assess the performances of CS in finding optimal parameters of JBC-PD control controller for the flexible manipulator control application. Throughout this work, all the simulations are carried out using Matlab/Simulink software.

II. THE FLEXIBLE MANIPULATOR SYSTEM

For the mathematical modeling of a flexible single-link manipulator, a finite element method and Lagrangian approach is utilized [9],[10],[11]. The development of the algorithm can be divided into three main steps:

1. FEM analysis
2. State space representation
3. Obtaining the result
The overall approach involves treating the link of the manipulator as an assemblage of \( n \) elements of length \( l \). For each of these elements the kinetic energy \( T_i \) and potential energy \( V_i \) are computed in terms of a suitably selected system of \( n \) generalized variables \( q \) and their rate of change \( \dot{q} \). In this study, a flexible single-link manipulator system is considered as shown in Fig. 1.

The link has been modelled as a pinned-free flexible beam. The pinned end of the flexible beam of length \( L \) is attached to the hub with inertia \( I_H \), where the input torque \( \tau(t) \) is applied at the hub by a motor and payload mass \( M_p \) is attached at the free end. \( E, I \) and \( \rho \) represent the Young Modulus, second moment of inertia and mass density per unit length of the flexible manipulator respectively. \( XY \) axis and \( X_0Y_0 \) axis represent the stationary and moving coordinate respectively. Both axes lie in a horizontal plane and all rotation occurs about a vertical axis. For a small angular displacement \( \theta(t) \) and small elastic deflection, \( w(x,t) \) the overall displacement \( y(x,t) \) of a point along the link at a distance \( x \) from the hub can be defined as a function of both the rigid rotation angle \( \theta(t) \) of the hub and flexible displacement \( w(x,t) \) of the beam measured from the line 0. Thus the total displacement \( y(x,t) \) is:

\[
y(x,t) = x\theta(t) + w(x,t)
\]

using the standard finite element method to solve dynamic problems, leads to the well-known equation:

\[
w_n(x, t) = N_n(s)Q_n(t)
\]

where \( N_n(s) \) and \( Q_n(t) \) represent the shape function and nodal displacement respectively.

Define, \( s = x - \sum_{i=1}^{n} l_i \), where \( l_i \) is the length of the \( i \)th element. The kinetic energy of an element can be expressed as follows:

\[
T_i = \frac{1}{2} [Q_n]^T[M_n][Q_n], \quad i = 1,2,\ldots,N
\]

and the potential energy due to the elasticity of the FE can be obtained as

\[
P_i = \frac{1}{2} \sum_{j=1}^{N} [Q_n]^T[K_n][Q_n],
\]

The total kinetic and potential energy can be written as:

\[
T = T_m + \sum_{i=1}^{N} T_i + T_p = \frac{1}{2} \dot{Q}^T M \dot{Q}
\]

Similarly, the potential energy due to the elasticity of the FE can be obtained as

\[
P = \sum_{i=1}^{N} P_i = \frac{1}{2} \dot{Q}^T \dot{K} \dot{Q}
\]

The Lagrangian of the link can be derived as followed:

\[
L = T - P = \frac{1}{2} \dot{Q}^T \dot{M} \dot{Q} - \frac{1}{2} \dot{Q}^T \dot{K} \dot{Q}
\]

A further detail on dynamic modeling can be found in[9][10][11]. By using Lagrange equation, the dynamic equations of a flexible manipulator can be derived utilizing the equation as followed: \( M'\ddot{Q} + D'\dot{Q} + K'Q = b'\tau \)

where \( M', K', Q' \) are mass matrix, rigidity matrix and general coordinates when substituting boundary conditions, respectively. \( b' = [1 \ 0 \ \cdots \ 0]^T \) \( \tau \) is input torque. \( D' \) is damping matrix and we choose the linear type Rayleigh damping and \( Q(t) \) is the nodal displacement given as

\[
Q(t) = [\theta \ w_1 \ \theta_1 \ \cdots \ w_n \ \theta_n]
\]

The \( M', D', K' \) matrices are of size \( m_1 \times m_1 \) and \( b' = [1 \ 0 \ \cdots \ 0]^T \) \( \tau \) has \( m_1 \times 1 \) size and \( m_1 = 2n+1. \) For simplicity \( D' = 0 \) and \( Q(0) = 0 \)

The state-space form of the equation of motion is

\[
\dot{v} = Av + Bu
\]

\[
y = Cv + Du
\]

where

\[
A = \begin{bmatrix}
0_{m_1} & I_{m_1} \\
\vdots & \vdots \\
-M^{-1}K & -M^{-1}D
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
0_{m_1} \\
M^{-1}e
\end{bmatrix}
\]

\[
C = \begin{bmatrix}
0_{m_1} & I_{m_1}
\end{bmatrix}
\]

\[
D = [0_{2m_1+1}]
\]

\( 0_{m_1} \) is an \( m_1 \times m_1 \) null matrix, \( I_{m_1} \) is an \( m_1 \times m_1 \) identity matrix, \( 0_{2m_1+1} \) is an \( m_1 \times 1 \) null vector, and the vector \( e \) is the first column of the identity matrix.

\[
u = [\theta \ u_1 \ \theta_1 \ \cdots \ u_{n+1} \ \theta_{n+1} \ \dot{\theta} \ \dot{u}_2 \ \dot{\theta}_2 \\
\vdots \ \ \ \ \ \ \ \ \ \ \ \ \ \dot{u}_{n+1} \ \dot{\theta}_{n+1}]
\]

Figure 1. Schematic diagram of the flexible manipulator system

Image 102x490 to 269x613
III. CUCKOO SEARCH ALGORITHM

Cuckoo search algorithm is a search algorithm developed by Xin-she Yang and Suash Deb 2009. The algorithm inspired by the breeding behavior of cuckoos. Cuckoo birds lay their eggs in other birds’ nests and relay on those birds for hosting the egg. Some of the other host birds discover that an egg is not their own, it might throw out the alien egg or just move to new locations elsewhere. A cuckoo might emulate the shape, color and size of the host eggs to protect their egg from being discovered. To increase the hatching probability of cuckoo birds own eggs, some of them might throw out other native eggs from the host nest. On the other hand, a hatched cuckoo chick will also throw other eggs away from nest to improve its feeding share [21][22].

The following three idealized rules used to describe the Cuckoo Search in simple way:

• Each cuckoo lays one egg at a time, and dumps it in a randomly chosen nest;
• The best nests with high quality of eggs (solutions) will carry over to the next generations;
• The number of available host nests is fixed, and a host can discover an alien egg with a probability $p_a$ [0, 1]. In this case, the host bird can either throw the egg away or abandon the nest so as to build a completely new nest in a new location.

**pseudo code of differential evolution**

n: population size , G: generation number
d: problem dimension ,
Begin
Objective function $f(x) , x = (x_1, x_2, x_3, \cdots x_d)^T$
Initial a population of $n$ host nests $x_i (i = 1, 2, 3, \cdots n)$
while ($t < $ MaxGeneration) or (stop criterion);
Get a cuckoo (say $i$) randomly and generate a new solution by Lévy flights;
Evaluate its fitness; $F_i$
Choose a nest among $n$ (say $j$) randomly;
if ($F_i < F_j$ )
Replace $j$ by the new solution;
end
Abandon a fraction (Pa) of worse nests [and build new ones at new locations via Lévy flights];
Keep the best solutions (or nests with quality solutions);
Rank the solutions and find the current best;
end while
Post process results and visualization;
End
The main procedures for application of Cuckoo Search is outlined in Fig. 2

IV. CS-JBC PD CONTROL STRUCTURE

A common strategy in the control of flexible manipulator systems involves the utilization of proportional and derivative (PD) feedback of collocated sensor signals, such as hub-angle and hub-velocity. Such a strategy is adopted through joint-based collocated (JBC) control.
The JBC controllers are capable of reducing the vibration at the end-point of the manipulator as compared to a response with open-loop bang-bang input torque. In this study, an intelligent proportional-derivative (PD)-like controller (JBC-PD) is used for rigid body motion. CS-JBC PD control system is adopted to position the flexible link to the desired angle of demand. The output command of JBC-PD control for the flexible manipulator control can be expressed as:

\[ U_m(t) = K_p e(t) - K_v \frac{d\theta(t)}{dt} \]

where \( U_m(t) \) is the control signal, \( K_p \) is proportional gain, \( K_v \) is the derivative gain, \( e(t) = \theta_R - \theta(t) \) is the error, \( \theta_R \) is the reference of angular displacement, and \( \theta(t) \) is the actual angular displacement.

In s-domain

\[ U_m(s) = K_p e(s) - K_v s \theta(s) \]

\[ U_m(s) = K_p \theta_R(s) - K_v s \theta(s) \]

The closed loop transfer function is therefore obtained as

\[ \theta(s) = \frac{K_p H(s)}{1 + K_v \left( s + \frac{K_p}{K_v} \right) H(s)} \]

where \( H(s) \) is the open loop transfer function. Here, CS is used to find the optimal value of JBC-PD parameters as shown in the Fig 3. The cost function will be optimized by CS is formulated based on the hub-angle error.

Figure 3. CS-tuned JBC PD control of flexible manipulator

V. IMPLEMENTATION AND RESULTS

To study the dynamic behavior of the flexible manipulator system, a computer program was written within MATLAB environment to simulate the state space matrices derived from the mathematical modeling done above. A thin aluminum alloy with the specifications shown in table 1 is considered [3].

<table>
<thead>
<tr>
<th>Components</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length ((l))</td>
<td>960 mm</td>
</tr>
<tr>
<td>Width ((w))</td>
<td>19.008 mm</td>
</tr>
<tr>
<td>Thickness ((h))</td>
<td>3.2004 mm</td>
</tr>
<tr>
<td>Mass density per unit volume (\rho)</td>
<td>2710 kg m(^{-3})</td>
</tr>
<tr>
<td>The second moment of inertia (I)</td>
<td>5.1924 \times 10^{-3} m(^4)</td>
</tr>
<tr>
<td>The young modulus (E)</td>
<td>71 \times 10^3 N m(^{-2})</td>
</tr>
<tr>
<td>The hub inertia (J_h)</td>
<td>5.86 \times 10^{-4} kg m(^2)</td>
</tr>
</tbody>
</table>

TABLE1: physical parameters of the flexible manipulator

For simplicity purposes, the effect of mass payload is neglected. Throughout this simulation, a bang-bang torque input with an amplitude ±0.3 Nm was applied at the hub of the manipulator as shown in Fig 4. The response of the flexible manipulator at the hub angle and end-point residual is monitored for duration of 3.0 seconds with sampling time 0.37 ms and is observed and recorded as shown in Fig 5(a),(b) and Fig 6(a),(b) in both time and frequency domain respectively. The first three resonant mode captured for end-point residual and hub angle response is at 11.59 Hz, 31.88 Hz and 58.93 Hz is compared with the experimental first three modes of vibration of the manipulator which obtained at 11.67 Hz, 36.96 Hz and 64.22 Hz respectively [3]. These, as noted, closely match with the corresponding experimental value with small percentage error that 0.6% for first mode, 13.7% for second mode and 8.2% for third mode.
VI. CONTROL SCHEMES

In this study, CS is used to find the optimal value of JBC-PD parameters using Matlab environment. The input reference used is bang-bang input with 1.3(rad) in magnitude, as shown in Fig 7. Bang-bang pattern movement input is chosen because it mirrors the nature of the task achieved by the manipulator. Besides, it is selected because it consists of one positive pulse and one negative pulse and is considered suitable to study the control performance using this signal [3].

The response of the flexible manipulator at the hub angle and end point residual are observed and recorded as shown in Fig 8 and Fig 9 in time domain. Both hub angle response and reference angle response is shown in fig 10.

From the work carried out it was found that satisfactory results were achieved with the following set of parameters:

- Population Initial Range for $K_p$: $[0; 3]$  
- Population Initial Range for $K_v$: $[0; 1]$  
- Number of Generations=40  
- the number of nests=10  
- parameter of Levy flights=1.5  
- Discovery rate of alien eggs=0.25

The resulted parameters is $K_p = 1.794653994705687, K_v = 0.557144537438670$.

It can be noted from these figures, that the system responses successfully track the desired hub angle. The manipulator reached the desired hub angle with the following result:

- Rise time for positive pulse $t_{ri}(s) = 0.58$  
- Settling time for positive pulse $t_{st}(s) = 0.84$  
- Maximum overshoot for positive pulse $M_{pi} = 0 \%$  
- Steady state error of positive pulse $e_{ss} = 0$  
- Decreasing time for negative pulse $t_{rd}(s) = 0.69$  
- Settling time for negative pulse $t_{sd}(s) = 0.94$  
- Maximum undershoot for negative pulse $M_{pd} = 0 \%$  
- Steady state error for negative pulse $e_{ss} = 0$  
- Maximum overshoot for the last stage of bang-bang input movement from $M_{pd} = 0 \%$  
- Steady state error for the last stage of input $e_{ss} = 0$

For the purpose of comparison, the hub angle response with PD controller designed using standard bacterial foraging algorithm (SBFA) and bacterial foraging algorithm with exponentially adaptable chemotactic step size (EABFA) approaches from [3] is shown in Table 2.

The numerical results presented in Table 2 show that CS-based control results in better time-domain hub-angle output performance, i.e. shorter rise time, decline time and settling time both in the positive and negative pulses.
TABLE 2. Time-domain hub-angle responses of the controllers

<table>
<thead>
<tr>
<th>control</th>
<th>CS-JBC PD</th>
<th>SBFA-JBC PD</th>
<th>EABFA-JBC PD</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_p$</td>
<td>1.7946</td>
<td>0.9394</td>
<td>1.8297</td>
</tr>
<tr>
<td>$K_v$</td>
<td>0.5571</td>
<td>0.4887</td>
<td>0.7325</td>
</tr>
<tr>
<td>$t_{ss}(s)$</td>
<td>0.58</td>
<td>0.8432</td>
<td>0.6957</td>
</tr>
<tr>
<td>$t_{rd}(s)$</td>
<td>0.48</td>
<td>1.4370</td>
<td>1.2703</td>
</tr>
<tr>
<td>$t_{rel}(s)$</td>
<td>0.69</td>
<td>1.1167</td>
<td>0.8877</td>
</tr>
<tr>
<td>$t_{sd}(s)$</td>
<td>0.94</td>
<td>1.7290</td>
<td>1.5004</td>
</tr>
</tbody>
</table>

VII. CONCLUSION
In this paper, CS algorithm have been adopted for tuning of JBC-PD, for hub-angle trajectory control. In this case CS algorithm has been used in controller design of a flexible manipulator in comparison with (SBFA) and EABFA design approaches. Simulation results have shown that the optimized PD controller of flexible manipulator system using was able to achieve better optimum value, and result in better time-domain hub-angle response in comparison to those with BFA and EABFA with no undershoot or overshoot, with shorter settling time and zero steady state error.

ACKNOWLEDGEMENT
The authors wish to thank the Ministry of Education (MOE) and Universiti Teknologi Malaysia (UTM) for providing the research grant and facilities. This research is supported using UTM Research University grant, Vote No. 04H17.

REFERENCES