Robust Falling-down Avoidance Control for Acrobat Robot Using Switching Controller

Y. Yashiro
Dept. of Mechanical and Control Eng.
Tokyo Institute of Technology
Tokyo, JAPAN
Email: yashiro@ac.ctrl.titech.ac.jp

M. Yamakita
Tokyo Institute of Technology
Tokyo, JAPAN
Email: yamakita@ctrl.titech.ac.jp

S. Hirano
The Institute of Physical and Chemical Research (RIKEN)
Nagoya, JAPAN

ZW. Luo
Dept. of Computer and Systems Eng.
Kobe University
Kobe, JAPAN

Abstract—In this paper, we propose methods of designing systematic controller including identification to stabilize the acrobat robot with reinforcement learning algorithm. We model the final stage of the robot at falling-down as a switched system with few contact conditions. For this model, we can systematically design a SDOF controller with for an unstable LTI system whose parameters are calculated by a closed loop identification. Considering the change of the dimension of the subsystem, we add a new subsystem with a reduced dimension to the switched system. We confirm the effectiveness of these methods with experiments using a real robot system.

Index Terms—Identification, Robot dynamics, Switched systems.

I. INTRODUCTION

Recently variable constraint systems are focused, and as a typical case of them, an acrobat robot with a horizontal bar was studied. For the generation of acrobatic motion with a horizontal bar, giant swing, taking off, somersault in the air and landing have been realized [3] [4] [5] [6].

And Falling-down avoidance control applying the reinforcement learning algorithm called Q-Learning to the robot was also realized [7] [8]. It is hard to design systematic controller for landing acrobot robot, because it has many contact conditions to the ground, and actuators of real robot system don’t move to the same way every time. In this point, it is effective to use reinforcement learning for acrobat robot. But this method needs unrealistic number of learning process, which leads to the difficulty to realize falling-down avoidance with the real robot system.

Therefore, we systematically design another controller to improve the robustness and reduce the number of learning process. The controller is designed for only the neighborhood of final desired posture of the robot, which has less contact conditions than those in Q-Learning. Since we have to model the neighborhood as an unstable switched system, we use a closed loop identification, and we design a switched dynamic output feedback (SDOF) controller to stabilize the system.

This method is, however, not originally considered the change of the number of the dimension when switching the subsystems. So we consider a the new subsystem with a reduced the dimension, and add to the switched system.

In this study, we confirm the effectiveness of the proposed method with experiments using a real robot system.

II. CONTROLLED SYSTEM

In this study, we use a humanoid type acrobat robot KHR-1 (produced by KONDO KAGAKU Co.) as a basic system. It has 17 servomotors, 2-axis gyro and 3-axis acceleration sensor on its head. It also has micro switches on the soles to sense their contact condition to the ground.

Fig. 1. KHR-1

A servo motor, KHR-786ICS is used at each joint.

<table>
<thead>
<tr>
<th>TABLE I</th>
<th>SPECIFICATION OF MOTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size</td>
<td>41×45×21[mm]</td>
</tr>
<tr>
<td>Mass</td>
<td>45[g]</td>
</tr>
<tr>
<td>Torque</td>
<td>8.1[kg/cm]</td>
</tr>
<tr>
<td>Speed</td>
<td>0.17[s/60[deg]]</td>
</tr>
</tbody>
</table>

- 2479 -
III. CONTROL LAW

Since the dynamics of the system change according to conditions to the ground, we model it as a switched system whose subsystems have some different dimensions. When we design a SDOF controller, we need subsystems for each contact condition, and subsystems for changing the dimension.

A. SDOF Controller

We define the switched system with N subsystems

\[ x(k+1) = A_i x(k) + B_i u(k) + B_w^i w(k), \]
\[ z(k) = C_i x(k) + D_i u(k) + D_w^i w(k), \]
\[ y(k) = C_i x(k) + D_w^i w(k) \quad (\alpha = \{1, 2, \ldots, N\}) \]

where \( x, z, y, u, w \) are state of system, controlled output, observed output, input, and disturbance. And subsystems can be switched \( i \rightarrow j (i, j \in \alpha) \) at arbitrary steps.

Then, if the LMI

\[
\begin{pmatrix}
X + X' - S_i & (\cdot) & (\cdot) & (\cdot) & (\cdot) \\
I + W - M_i & Y + Y' - N_i & (\cdot) & (\cdot) & (\cdot) \\
0 & 0 & \gamma I & (\cdot) & (\cdot) \\
A_iX + B_iL_i & A_i + B_iR_iC_i & B_w^i + B_iR_iD_w^i & S_j & (\cdot) \\
Q_i & YA_i + FC_i & F_iD_w^i + YB_w^i & M_j & N_j \\
C_i^T X + D_i^T L_i & C_i^T + D_i^T R_iC_i & D_i^T w_i + D_i^T R_iD_w^i & 0 & 0 & \gamma I
\end{pmatrix} > 0 \quad ((i, j) \in \alpha \times \alpha)
\]

has the solution \( (S_i, M_i, N_i, X, Y, W, L_i, F_i, Q_i, R_i, \gamma) \), SDOF controller is given as [1].

\[
\eta(k+1) = A_i^\gamma \eta(k) + B_i^\gamma y(k),
\]
\[ u(k) = C_i^\gamma \eta(k) + D_i^\gamma y(k) \quad (3)
\]

\[
A_i^\gamma = V^{-1} [Q_i - Y (A_i + B_i R_i C_i) X - V B_i^2 C_i X - YB_i^2 C_i U U^{-1},
B_i^\gamma = V^{-1} (F_i - Y B_i^2 R_i),
C_i^\gamma = (L_i - R_i C_i X) U^{-1},
D_i^\gamma = R_i, VU = W - YX
\]

B. \( \gamma \)-performance

The controller designed by (3) is said to have the \( \gamma \)-performance, and satisfy

\[
\gamma^{-1} \sum_{k=0}^{\infty} z(k)^2 < \gamma \sum_{k=0}^{\infty} w(k)^2, \forall \sum_{k=0}^{\infty} w(k)^2 > 0.
\]

This condition means that the \( L_2 \) gain of the closed loop system is less than or equal to \( \gamma \). Therefore, when we design the SDOF controller, we add the condition of minimizing \( \gamma \) to LMI solver.

C. Subsystem with Reduced Dimension

When switching from the subsystem with higher dimension to that with lower dimension, we assume external force \( \lambda(k) \) and

\[
x(k+1) = A(x) + B u(k) + B_w w(k) + E \lambda(k)
\]
\[ E^T x(k+1) = 0 \]
are satisfied where \( E x(k) = 0 \) stands for constraints. By eliminating \( \lambda \) from two equations, we have

\[
x(k+1) = (I - E (E^T E)^{-1} E^T ) A(x) + (I - E (E^T E)^{-1} E^T ) B u(k)
\]
\[ + (I - E (E^T E)^{-1} E^T ) B_w w(k)
\]
\[ = A' (x) + B' u(k) + B_{w'} w(k).
\]

We get new subsystem \( (A', B', B_{w'}, C) \), and when the constraint is satisfied, 1 step of this subsystem is inserted between the two subsystems.

IV. MODELING WITH IDENTIFICATION

Since we control the robot with the input to motor angle, not torque, and need to get unstable model from the data, from the stabilized closed loop system, we use a closed loop identification method.

A. Parameter of Robot

As shown in Fig.2, we consider the robot 9 links body in this study. The posture of the robot is defined as

\[
\Theta = [\theta_3, \theta_4, \theta_5, \theta_9, \theta_{31}, \theta_{41}, \theta_{51}, \theta_{91}]^{T} \quad (8)
\]

The position of C.O.G of the robot in \( \Sigma_0 \) coordinate is calculated with the posion of C.O.G, weight, and relative angle of each link. The origin of \( \Sigma_0 \) coordinate is the center of the box made by toes and heels. Thus the position and velocity of the C.O.G of the robot are expressed with vectors

\[
\]

Input and output for identification and controller are

\[
u = [u_1, u_2, u_3] = [\theta_3, \theta_4, \theta_5],
\]
\[y = [y_1, y_2] = [G_X, G_Z] - y_{des},
\]

where \( y_{des} \) is the final desired position.

B. Closed Loop Identification

We use exogenous input \( [r_1, r_2]^T \), and combined output \( p = [y, u]^T \) in the closed loop shown in Fig.3. We get the innovation model expressed in (11) with the identification.

\[
x(k+1) = A x(k) + [B_1 \quad B_2] \begin{bmatrix} r_1(k) \\ r_2(k) \end{bmatrix} + \begin{bmatrix} K_1 & K_2 \end{bmatrix} \begin{bmatrix} e_1(k) \\ e_2(k) \end{bmatrix},
\]
\[ p(k) = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x(k) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \end{bmatrix} \begin{bmatrix} r_1(k) \\ r_2(k) \end{bmatrix} + \begin{bmatrix} e_1(k) \\ e_2(k) \end{bmatrix}
\]

(11)
Transforming with $r_1 = e_2 = 0$, we get the model of the systems as

$$x(k+1) = [A - B_2D_{22}^{-1}C_2]x(k) + B_2D_{22}^{-1}u(k) + K_1e_1(k),$$
$$y(k) = C_1x(k) + e_1(k).$$

(12)

We use ORT method, which is one of the subspace identification, to calculate matrices of the model [1]. The dimensions of the matrices are the sum of the dimension of the plant and the dimension of the controller. Therefore we need to lower the dimensions using SR algorithm, including deriving grammian of the system and SVD.

1) ORT method: ORT method gives the good result when the data for identification have colored noise, because this method divides the data into deterministic, and stochastic parts with orthogonal decomposition.

Because of this, the model of the system is expressed in the combination of deterministic subsystem, and stochastic subsystems as

$$\begin{bmatrix} x(k+1) \\ x_s(k+1) \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & A_s \end{bmatrix} \begin{bmatrix} x(k) \\ x_s(k) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(k) + \begin{bmatrix} 0 \\ K_s \end{bmatrix} e(k),$$
$$y(k) = \begin{bmatrix} C \\ C_s \end{bmatrix} \begin{bmatrix} x(k) \\ x_s(k) \end{bmatrix} + Du(k)+e(k).$$

(13)

If the dynamics of two subsystems coincide, the model can be expressed in (11).

2) SR algorithm for unstable system: SR algorithm use the grammian of the system. When it is unstable, controllable and observable grammians $P, Q$ are given with Riccati equations and Lyapnov equation as

$$X = A^T(X - XB[I_m + B^T XB]^{-1}B^T X)A,$$
$$Y = A(Y - YC^T[I_p + CYC^T]^{-1}CY)A^T$$

$$F = -(I_m + B^T XB)^{-1}B^T X A,$$
$$L = -AYC^T(I_p + CYC^T)^{-1}$$

$$P = (A + BF)P(A + BF)^T + B(I_m + B^T XB)^{-1}B^T,$$
$$Q = (A + LC)^T Q(A + LC) + C^T(I_p + CYC^T)^{-1} C.$$  

(16)

And calculate the SVD of the product $SR$, where $P = S^T S, Q = R^T R$. We compare to the size of each singular value of SVD, and we can deside the number to which we lower the dimension of the system.

V. EXPERIMENT OF IDENTIFICATION

We model the subsystems for each contact condition with ORT method. For this method, we use SDOF controller designed with matrices calculated MATLAB command n4sid. We get the data for n4sid with the real robot system under stable final desired posture, so the system model of these matrices is also stable.

In ORT method, we use random $r_1$ and random integer $r_2$,  
$$| r_1 | \leq 0.005, \quad | r_2 | \leq 2$$  

(17)
as exogenous input. The numbers of data for ORT method in each subsystem are 1627 for normal contact, and 1123 for tip toe contact. The size of calculated matrices is $6 + 6 = 12$ : the size of the stable system + the size of SDOF controller. Considering the existence of feet bottom angle, we choose and lower the dimension of normal contact condition to 6, and tip toe contact condition to 8. The eigenvalues of two subsystems are shown in Fig.4 and Fig.5. Some eigenvalues are out of unit circle, so the subsystems modeled with closed loop identification are out-target unstable systems. Eigenvalues of deterministic and stochastic subsystems have some similarities, we express equations in (11).

VI. EXPERIMENT OF SDOF CONTROLLER

We apply SDOF controller designed with closed loop identification to the real robot system.
A. Condition

- Sampling interval of robot is 75 [ms].
- Subsystems of the posture we have to stabilize are
  mode1: normal contact condition
  mode2: tiptoe contact condition
  mode3: reducing dimension condition
and switching rule among subsystems is shown in Fig.6. When mode2, we add condition with 2 step continuous reverse of the robot because of the delay of input to servo motor.
- We let the robot fall-down with the 2 steps,
  \[ \Theta = \Theta_{final} + [-9, 0, 0, -9, 0, 0, 0, 0] \text{[deg]} \]  
  and after that, SDOF controller starts to inject the control input to the servo motors. SDOF controller stops when position and velocity of C.O.G reaches the neighborhood of \( y_{des} \).
- The final desired posture is
  \[ \Theta_{final} = [-55, 80, -28, 15, -55, 80, -28, -15] \text{[deg]} \]
  \[ y_{des} = [-0.003, 0.176] \text{[m]} \]  
  (19)

B. Result

The result is shown in Fig.7, where position and velocity of the C.O.G and controller mode number are shown, and Fig.8, where input and servo motor angles are shown. They show that the robot avoids falling-down and C.O.G and servo motor angles nearly converge with the final desired posture. Fig.9 shows the motion of the real robot system.

VII. EXPERIMENT WITH MOVING PATTERN

The main purpose of SDOF controller is to reduce the number of reinforcement learning process. Sufficient Q-Learning process can design the moving pattern with 4 posture from initial posture to final desired posture [8]. But with real robot system, the moving pattern is not perfect because of the short number of learning process, and robot fall-down.

So we use SDOF controller at the end of the moving pattern with 4 posture which cannot avoid the robot from falling-down, and check the effectiveness the combination of SODF controller and Q-Learning.

A. Condition

Most of the experimental conditions are the same in VI.A, but in addition to 2 step falling-down, we let the robot move as SDOF controller is low gained as

\[ u \leftarrow 0.14u \text{ (mode = 1)} \]
\[ u \leftarrow 0.20u \text{ (mode = 2)} \]
\[ u \leftarrow 0.24u \text{ (mode = 3)} \]  
(20)
because the size of the control input of the original controller designed by the LMI solver which minimizes \( \gamma \) is too large for the real robot system to keep links and motors good and unbreakable. We decide the levels of lowering gain and check the stability of the controller with a simulater, which has the same conditions of the real system.
During moving, SDOF controller starts if starting condition (22) is satisfied.

\[
\Theta(k) = \begin{cases} 
[0, 4, 7, -25, -25, 60, -60, 15] & \text{if } k=1,2,3,4,5 \\
[-40, 60, -20, -25, -28, 60, -47, 25] & \text{if } k=6,7,8 \\
[-37, 70, -40, -15, -37, 60, -35, 15] & \text{if } k=9,10 \\
[-50, 80, -29, -15, -50, 80, -29, 15] & \text{if } k=11,12,13
\end{cases}
\]

During moving, SDOF controller starts if starting condition (22) is satisfied.

\[
\begin{cases} 
k \geq 11 \\
| y_1 | \geq 0.02
\end{cases}
\]
condition to switch subsystems, and starting condition of SDOF controller.

And we also need to make sure the reduction of reinforcement learning process by the combination with systematic controller.

REFERENCES