A RECEDING HORIZON CONTROL FOR LIFTING PING-PONG BALL

Sumiko Majima and Keii Chou

Graduate School of Systems and Information Engineering,
University of Tsukuba, Tsukuba, Ibaraki, Japan

Abstract: This paper proposes a control system for lifting a ping-pong ball without the use of a camera. The system measures the time of collision between a plate and a ball with a microphone. The control system must then be able to direct both the position and the velocity of a motor for each subsequent collision, even if the model contains uncertainty or modeling errors. To achieve this objective, an optimal control is modified using receding horizon control. The effectiveness of the proposed control method is verified by performing simulations on the lifting system.

Keywords: Optimal control, Constrain satisfaction problem, Modelling errors, Feedback control, Hybrid, Intervals

1. INTRODUCTION

Various visual servo systems have been developed for controlling flying objects. For instance, Andersson (1989) built a system to play ping-pong, which employed multiple cameras to track the ball in three dimensions. Corke (1996) developed a visual feedforward controller for an eye-in-hand manipulator to track a ping-pong ball thrown across the system's field of view. Both of these systems perform well, but they are complex and expensive. A cheaper system without cameras can be designed if the goal is restricted to simply lifting ping-pong balls in one dimension. This paper proposes a system for lifting a ball that uses a microphone to measure the collision time.

In the system, the position and the speed of the motor at each collision should be controlled to satisfy the desired values. The initial time is the time at which a collision occurs and the final time is the time at which the next collision will occur. The final time is estimated by predicting the motion of the flying ball. This information is then used to direct the angular position and velocity of the motor to prepare for the next collision. This problem can be solved using an optimal control. A classical optimal control, though, demands an exact description of the model. But in most cases, the model includes an uncertainty or modeling error; these types of errors will cause the actual trajectory to deviate from the desired trajectory estimated by nominal model. Therefore, the control system must provide satisfactory performance in the face of modeling errors. A version of feedback control should be introduced to improve the robustness of the controller. Receding horizon control is a feedback method in which the control inputs are obtained by solving the finite horizon optimal control problem based on measured data at each sampling interval. (Kwon et. al, 1983, Ohtsuka, 1999). By modifying the receding horizon control, this paper proposes a control system for leading state variables to desired values at each interval even if the controlled model contains uncertainty or modeling errors. The proposed system is then applied to a system designed to lift a ping-pong ball without a camera.

This paper is organized as follows. Section 2 describes the equipment developed for lifting a ping-pong ball and presents the basic experimental results. Section 3 introduces the optimal control system that has been modified to perform well even if the motor model contains uncertainty or modeling errors. The proposed receding horizon control system is then applied to a control system for lifting a ping-pong ball in Section 4. In order to lift the ball at the desired height despite errors in the collision coefficient, another control method is inserted in the system. Section 5 shows simulation results and a conclusion is provided in Section 6.

2. PING-PONG BALL LIFTING CONTROL

2.1 Equipment

A lifting machine was designed that consists of three links, a DC motor, and a plate. A microphone is attached to the plate to determine the collision times. The length of Link 1 is L_1 and the length of Link 2 is L_2. The third link is supported by a guide and the plate moves vertically.

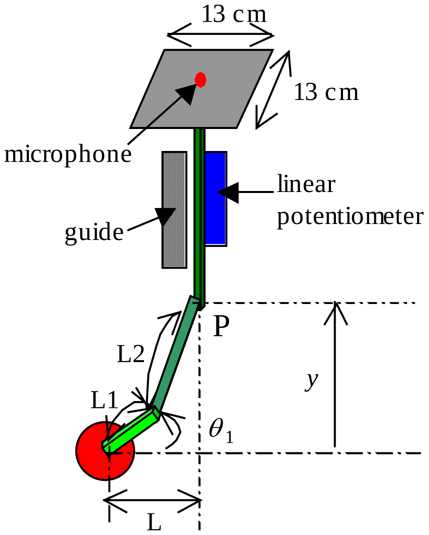

Fig. 1 Schematic of the ping-pong-ball lifting machine

The perpendicular distance between the center of the motor and the center of the plate is *L*. A linear potentiometer is used to measure the distance that plate *y* moves. This distance *y*, which is equivalent to the displacement of P shown in Fig. 1, is calculated using a function of the motor's angular position θ_1:

$$y = L_1 \sin\theta_1 + L_2 \sin(\cos^{-1}(\alpha - \beta\cos\theta_1)) \quad (1)$$
$$\begin{cases} \alpha = L / L_2 \\ \beta = L_1 / L_2 \end{cases}$$

The dimensions of the equipment shown in Fig. 1 is: $L=0.1\ m$, $L_2=0.2\ m$ and $L_1=0.025\ m$. The plate moves between *0* and *0.0579 m*.

Fig. 2 shows a voltage time response for the microphone amplifier when a ball is dropped on the plate. The ball continued to bounce. The voltage peaks indicate each time that the ball collided with the plate. With time, the height of the ball decreased due to energy losses, i.e., the collision coefficient is less than 1. In addition, the interval between subsequent collisions decreased, as well. Fig. 2 confirms this characteristic. In the figure, the time-intervals between peak voltages gradually decreased. A collision coefficient *e* of 0.65 was estimated from these measured intervals.

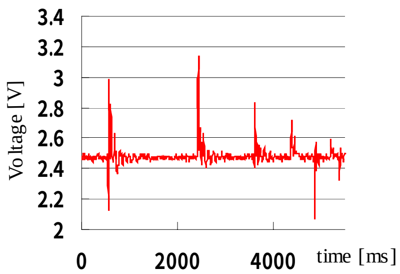

Fig. 2 Microphone time response for the bouncing ball

2.2 Open Control

After a ball is initially dropped on a plate, the ball continues to bounce on the plate again and again, and its peak height decreases gradually. The peak height, however, can become constant if the plate hits the ball at a certain speed every time the ball collides with the plate. This action is equivalent to lifting the ping-pong ball. An easy method for lifting the ball is to drive the motor at a constant speed specifically adjusted for the bouncing ball. This constant speed was determined by trial and error. Experiments were then carried out at this speed.

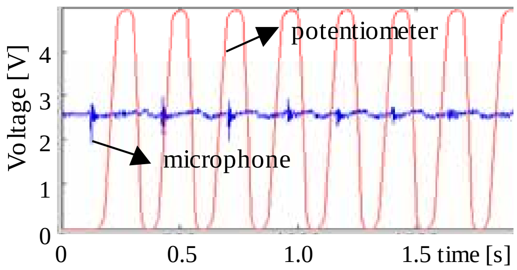

Fig. 3 Voltage time responses during ball lifting

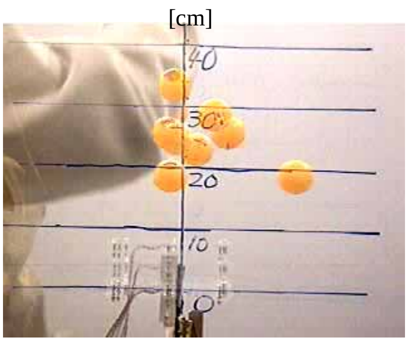

Fig. 4 Peak height after each lifting

Fig. 3 shows the voltage time responses for the potentiometer and the microphone during ball lifting. Fig. 4 displays the peak height for every bounce. These results indicate that the ball bounced at a constant interval and its peak height remained fairly constant. Therefore, the ping-pong ball can be bounced at a constant peak height. However, this experiment can only be performed at fixed motor speed, and its peak height is determined by the speed. Overall, it is difficult to lift a ping-pong ball at any constant height using this method. In order to lift a ball at any height, the control system described in Section 3 will be developed.

3. RECENDING HORIZON CONTROL

In order to lift a ball at various heights, the speed at which the plate contacts the ball must be varied. Furthermore, the plate should hit the ball at the same location for each collision. Optimal control is used to achieve these objectives. The initial time is the time at which a collision occurs and the final time is the time at which the next collision will occur. The final time is calculated from the model of lifting the ball. The final position and the desired final speed of the motor are given; the initial position and velocity of the motor are measured and known. Then, the problem is solved by an optimal control problem with fixed terminal state.

3.1 Optimal Control with Fixed Terminal State

The dynamics of the motor are approximated by the following transfer function. Input u is the voltage across the motor and output y is the motor's angular position.

$$G(s) = a/s^2 \tag{2}$$

The initial constraints are:

$$y(0) = y_{0,} \quad \dot{y}(0) = v_0 \tag{3}$$

and the final constraints are:

$$y(T) = y_T, \quad \dot{y}(T) = v_T \tag{4}$$

The control input at each time t is the value that minimizes the performance index:

$$\int_0^T u^2 \, dt \tag{5}$$

The input for the minimum performance index can be written as:

$$u = -(C_1 t + C_2)$$
$$\begin{cases} C_1 = -(6v_0T + 6v_TT + 12y_0 - 12y_T)/(a^2T^3) \\ C_2 = -(4v_0T + 2v_TT + 6y_0 - 6y_T)/(a^2T^2) \end{cases} \tag{6}$$

The angular position y(t) and the velocity $\dot{y}(t)$ can be expressed by:

$$y = -(1/6)a^2C_1t^3 - (1/2)a^2C_2t^2 + v_0t + y_0 \tag{7}$$

$$\dot{y} = -(1/2)a^2C_1t^2 - a^2C_2t + v_0 \tag{8}$$

The inputs described in Equation (6) accelerate the motor from its initial constraint $y(0) = y_0$, $\dot{y}(0) = v_0$ to its final constraint $y(T) = y_T$, $\dot{y}(T) = v_T$. In most cases, however, the plant model used to solve the problem contains uncertainty or modeling errors. These types of errors will cause the actual trajectory to deviate from the desired trajectory represented in Equations (7) and (8), and will prevent the motor from successfully meeting the desired final constraint. A version of feedback control should be introduced to improve the robustness of the controller. Receding horizon control is a feedback method in which the control inputs are obtained by solving the finite horizon optimal control problem based on measured data at each sampling interval. New inputs are calculated based upon the measured data and are set as the current control input. This calculation procedure is performed continuously, yielding a feedback control. The prediction horizon T, which is equivalent to the final time T in Equation (5), is constant. The horizon moves with time as shown in Fig. 5(a). This method, though, is not very effective for making variables converge to their desired values at a designated time. In this paper, the horizon value decreases with time t and is updated after every interval dh, as shown in Fig. 5(b). Using this method, the performance index at the final interval can be calculated as:

$$\int_0^{dh} u^2 \, dt \tag{9}$$

If the update interval dh is equal to the sampling time t_s, dh is very small and the calculated parameters C_1 and C_2 in Equation (6) tend to be exceptionally large at the final intervals. To avoid these large values, dh is restricted to be larger than t_s. The final time T is divided into N parts and T/N=dh is called the update interval.

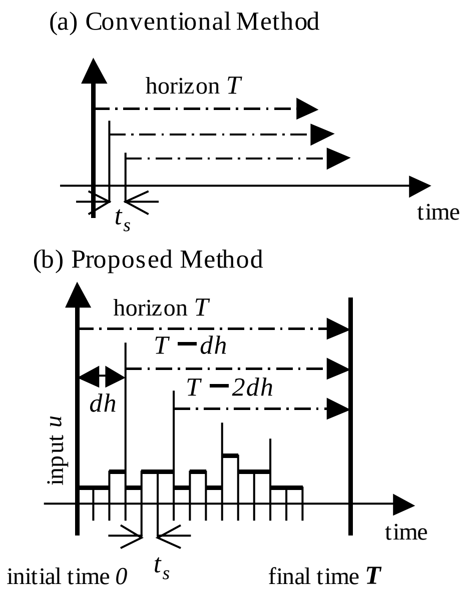

Fig. 5 Method for Changing Horizon

3.2 Algorithm of Proposed Receding Horizon Control

The procedure employed for obtaining inputs is as follows.

1. Set the initial time as *0*. The inputs are calculated using Equation (6) with a horizon = T and initial constraints $y(0) = y_0$ $\dot{y}(0) = v_0$. The calculated control inputs are sent to the motor at each sampling time t_s to drive the motor.
2. At time dh, the initial constraints are changed to $y(dh) = y_{01}$ and $\dot{y}(dh) = v_{01}$. Data y_{01} and v_{01} are then measured. Horizon T is replaced by T-dh. The inputs

are again calculated from Equation (6) using these updated values.

3. Using the same method, the horizon and initial constraints are updated at each update interval *dh*. The inputs are then calculated from these new values.

Repeating the input calculations with the measured data and the new horizon at each update interval will reduce the effects of the modeling error.

3.3 Performance of the Proposed RH Control

In this section, the performance of the proposed receding horizon control system is compared with a classical optimal control system. The motor is represented by the following transfer function:

$$G(s) = \frac{a}{s(s+\alpha)} \quad (9)$$

where $\alpha = 0.3$. The inputs for the classical optimal control system are determined from Equation (6), which is derived from the transfer function in Equation (2). The parameter α is not considered when the optimal control problem is solved.

Fig. 6 shows time response of a position y and a velocity $\dot{y}$ that satisfy the conditions:

$$y(0) = 6.28,\ \dot{y}(0) = 3.14*5.3$$
$$y(T = 0.5) = 3.14*4,\ \dot{y}(T = 0.5) = 3.14*5.3$$

,where the update interval *dh* is *50* ms, the division number *N* is *10* and the sampling time t_s is 5 ms. In the figure, the proposed control system is labeled "RH control." The velocity of the proposed system reaches the final desired values, while the ordinal optimal control system deviates.

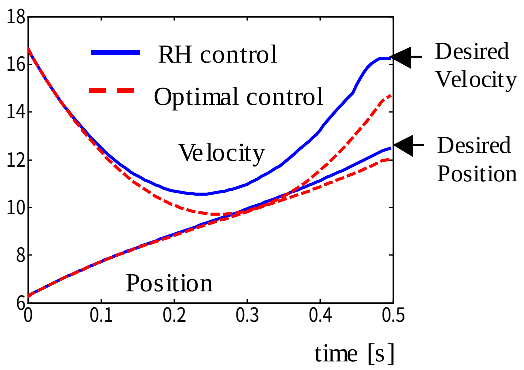

Fig. 6 Compare RH control with classical optimal control

Next, the performance of the proposed receding horizon control is investigated in terms of the division number *N*. For this experiment, the constraints are set to

$$y(0) = 0,\ \dot{y}(0) = 0$$
$$y(T = 0.5) = 3.14*4,\ \dot{y}(T = 0.5) = 3.14*5.3$$

Fig. 7 shows time responses of the motor velocity and the motor inputs. The figure demonstrates that the division number *N* affects the performance of the control, but the difference in the final error between *N*=4 and *N*=10 is slight. The slope of inputs, i.e., the parameter C_1 in Equation (6) depends on the division number *N*. The slope at the final interval is very great when *N*=10. Then, we set *N*=4 on simulations in Section 4 and Section 5.

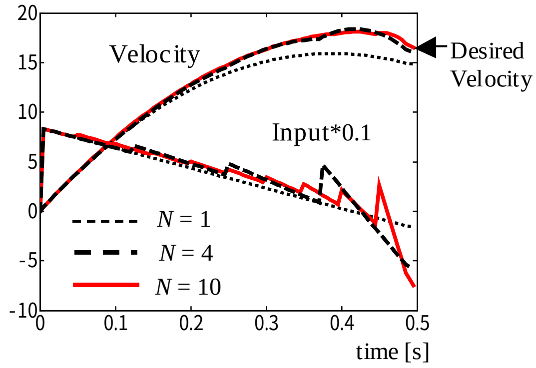

Fig. 7 Comparison of the quasi-RH performance for different division numbers *N*

4. CONTROL SYSTEM FOR LIFTING A BALL

A control system for lifting the ball consists of two controllers as shown in Fig. 8. The one controller (*Controller 1*) determines three desired values for lifting a ball at the desired height H_r: the time T_{ref} at which the next collision will occur, the height of the plate Y_{ref} at which the plate contacts the ball, and the velocity of the plate V_{ref} at which the plate hits the ball. Then the motor should be driven to achieve these three desired values using the other controller (*Controller 2*). The receding horizon controller described in Section 3 can be used for the *Controller 2*. In this section, a method for *Controller 1* is proposed.

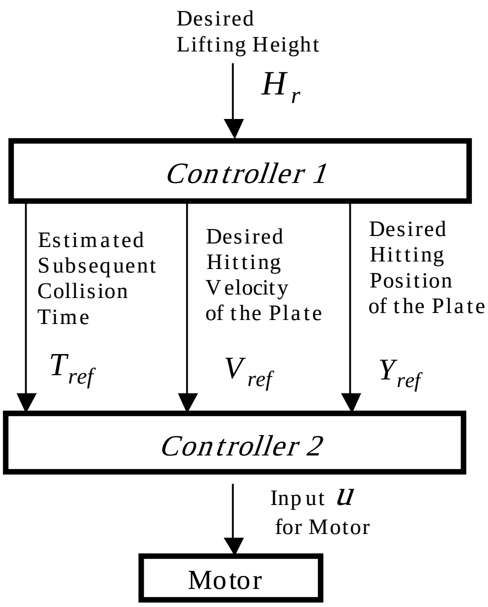

Fig. 8 Control System for Lifting a Ball

At the k-th collision, the *Controller 1* determines the next subsequent collision time $T_{ref}(k+1)$, the next desired hitting velocity $V_{ref}(k+1)$, and the next desired hitting height $Y_{ref}(k+1)$ with the following method.

First, the next desired hitting height $Y_{ref}(k+1)$ of the plate is determined. When the plate passes through the center of the range of its up-and-down movement, it has the maximum velocity; this center height Y_{pref} is the most efficient hitting height to lift the ball. Hence, we set $Y_{ref}(k+1)$ to Y_{pref}.

$$Y_{ref}(k+1) = Y_{pref} \tag{10}$$

Second, the next subsequent collision time $T_{ref}(k+1)$ is estimated. At the k-th collision, the plate hits the ball at a height $Y_p(k)$ and after that the ball moves up with a velocity $V_{ba}(k)$. And then, the ball falls down and collides with the plate at the desired height Y_{pref}. The interval between two succeeding collision, i.e., the next subsequent collision time $T_{ref}(k+1)$ is estimated by the following equation

$$T_{ref}(k+1) = \left\{ V_{ba}(k) + \sqrt{V_{ba}^2(k) - 2g(Y_{pref} - Y_p(k))} \right\} / g \tag{11}$$

Last, the next desired hitting velocity $V_{ref}(k+1)$ is determined using the law of conservation of momentum. The ball velocity $V_{ba}(k)$ immediately after the collision is estimated as

$$V_{ba}(k) = \frac{(1+e)MV_{Lf}(k) + (m - Me)V_{bf}(k)}{m+M} \tag{12}$$

where

- e: collision coefficient
- m: mass of the ball
- M: mass of the plate
- $V_{Lf}(k)$: velocity of the plate immediately before the k-th collision
- $V_{bf}(k)$: estimated velocity of the ball immediately before the k-th collision.

Because the plate is connected to the link and the motor, the motion of the plate may not be affected by the collision. Hence, we assume that $M << m$ and use the following equation instead of equation (12).

$$V_{ba}(k) = (1+e)V_{Lf}(k) - eV_{bf}(k) \tag{13}$$

In the case that the maximum height of the ball is the desired height H_r, the ball velocity $V_{ba}(k+1)$ immediately after the $(k+1)$-th collision should be

$$V_{ba}(k+1) = \sqrt{2g(H_r - Y_{pref})} \tag{14}$$

Then the next desired hitting velocity $V_{ref}(k+1)$ of the plate to achieve this desired ball velocity is obtained from equations (13) and (14).

$$V_{ref}(k+1) = \{\sqrt{2g(H_r - Y_{pref})} + eV_{bf}(k+1)\}/(1+e) \tag{15}$$

The law of conservation of momentum is true only when no driving force is applied during the collision. In the equipment for lifting the ball, the plate hits the ball and the driving force is applied at each collision; hence equation (12) is not accurate to represent the real collision phenomena in this case. We assume that this inaccuracy can be expressed through errors in the estimated collision coefficient e. The errors degrade the performance of the controller and will inhibit lifting the ball at the desired height. Hence, we insert a feedback controller in the *Controller 1*.

The real subsequent collision time $T(k)$ depends on the real lifting height and the time $T(k)$ can be measured using the microphone. On the other hand, the desired subsequent collision time T_{des} for lifting the ball at the desired height H_r is

$$T_{des} = 2\sqrt{2g(H_r - Y_{ref})} / g \tag{16}$$

The desired hitting velocity $V_{ref}(k+1)$ is modified using the following PI controller for decreasing the deviation between T_{des} and $T(k)$:

$$V_{ref}(k+1) = K_p e_T(k) + K_i \sum_j e_T(j) \quad \text{where} \quad e_T(k) = T_{des} - T(k) \tag{17}$$

The modeling errors also shift the hitting height from the desired height Y_{pref}, hence the desired hitting height $Y_{ref}(k+1)$ is also modified using the following PI controller:

$$Y_{ref}(k+1) = K_{py} e_y(k) + K_{iy} \sum_j e_y(j) \quad \text{where} \quad e_y(k) = Y_{pref} - Y_p(k) \tag{18}$$

The algorithm of *Controller 1* is summarized as follows:

1) The next subsequent collision time $T_{ref}(k+1)$ is estimated from equation (11):

$$T_{ref}(k+1) = \left\{ V_{ba}(k) + \sqrt{V_{ba}^2(k) - 2g(Y_{pref} - Y_p(k))} \right\} / g \tag{19}$$

2) The next desired hitting velocity $V_{ref}(k+1)$ is obtained from both the feedforward input described in equation (15) and the collision time feedback input described in equation (17):

$$V_{ref}(k+1) = \{\sqrt{2g(H_r - Y_{pref})} + eV_{bf}(k+1)\}/(1+e) + K_p e_T(k) + K_i \sum_j e_T(j) \tag{20}$$

3) The next desired hitting height $Y_{ref}(k+1)$ is obtained from both the feedforward input described in equation (10) and the hitting height feedback input described in equation (18):

$$Y_{ref}(k+1) = Y_{pref} + K_{py} e_y(k) + K_{iy} \sum_j e_y(j) \tag{21}$$

The determined height $Y_{ref}(k+1)$ and the velocity $V_{ref}(k+1)$ of the plate are transferred into the hitting position $M_{ref}(k+1)$ and the velocity $V_{mref}(k+1)$ of the motor using equation(1). The motor is driven to achieve the determined velocity $V_{mref}(k+1)$ and the position $M_{ref}(k+1)$ at the estimated time $T_{ref}(k+1)$ using *Controller 2*, which is proposed in Section 3.

5. SIMULATION OF LIFTING A BALL

The lifting of a ping-pong ball controlled using the proposed control system was simulated under the following conditions.

1) The ball was held at an initial height of 0.65 *m* and released at time 0. Initially, the motor was stationary. After the first collision between the ball and the plate, the motor began to move.
2) The nominal collision coefficient *e* was set to 0.65 and the error in the collision coefficient is set to -0.02, i.e., the real collision coefficient was set to 0.63.
3) In *Control 1*, the gains of PI controller were set as K_p=30, K_i=3, K_{py}=10 and K_{iy} = 5.
4) In *Controller 2*, the division number *N* for the receding horizon control was always set to 4. The motor was represented by the transfer function in equation (9), where $\alpha = 0.3$.
5) The desired heights H_r of the lifting ball were set as follows:

$$H_r = 0.25\,[m] \quad t \le 8\,[s]$$
$$H_r = 0.4\,[m] \quad t > 8\,[s]$$

In the first simulation, *Control 1* involves no feedback controller, that is, K_p=0, K_i=0, K_{py} =0 and K_{iy} = 0. The trajectory of the plate and the ball are shown in Fig. 9. The ball bounced rhythmically and the maximum height of the ball changes rapidly when the desired height H_r is switched from 0.25[m] to 0.4[m]. However, the maximum height does not reach the desired values because of the errors in collision coefficient *e*.

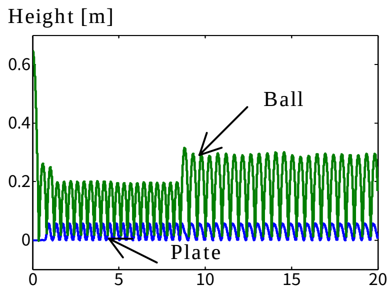

Fig. 9 Time responses without feedback controller

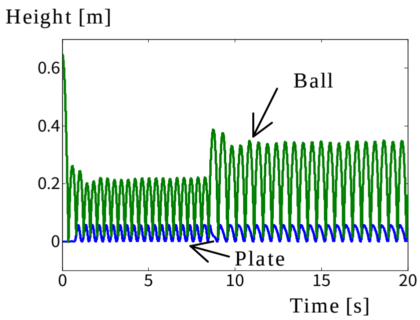

Fig. 10 Time responses with only collision time feedback controller

In the second simulation, *Control 1* involves only the collision time feedback controller, that is, K_p=30, K_i=3, K_{py} =0 and K_{iy} = 0. Fig. 10 shows that the heights of the ball approaches to the desired heights but not reaches them. In this case, the collision height of the plate is much less than the desired height Y_{pref}, hence the hitting velocity of the plate is less than the determined value $V_{ref}(k+1)$. This result indicates the significance of the hitting position feedback.

In the last simulation, *Control 1* involves both the hitting position feedback and the collision time feedback. Fig. 11 shows that the ball height changes rapidly when the desired height H_r is switched from 0.25[m] to 0.4[m] and that the maximum height of the ball reaches the desired values despite the modeling errors.

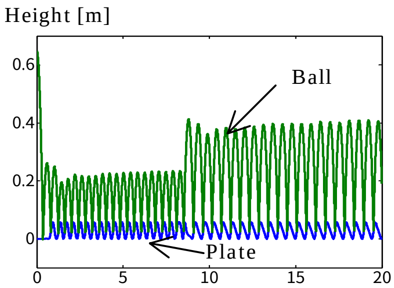

Fig. 11 Time responses with feedback controller

6. CONCLUSIONS

This paper proposed a control system for directing variables towards their desired values at each time interval even if the model contains uncertainty or modeling errors. To achieve this objective, an optimal control system was modified based on receding horizon control. The proposed control system was then applied to a system designed to lift a ping-pong ball without the use of a camera. For lifting the ball at the desired height, another controller was inserted in the system. The role of this controller was to estimate the subsequent collision time and to determine the next hitting position and the next hitting velocity at the estimated time. The effectiveness of the proposed control system was verified by performing simulations.

REFERENCES

Andersson, R.L (1989), Aggressive Trajectory Generator for a Robot Ping-pong Player. *IEEE Control Systems Magazine*, Vol. 9, No. 2, pp. 15-21

Croke, P.I. (1996), Dynamic Effects in Visual Closed-Loop Systems. *IEEE Trans. On Robotics and Automation*, Vol. 12-5, pp.671-684

Kwon, W.H, Bruckstein, A.N, and Kaitath, T (1983), Stabilizing state-feedback design via the moving horizon method. *International Journal of Contol*, Vol. 37-3, pp. 631-643.

Ohtsuka, T. (1999), Time-variant receding-horizon control of nonlinear systems, *Journal of Guidance, Control and Dynamics*, Vol. 21, no. 1, pp. 174-176.