Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Model-Following Control of Nonlinear Systems based on Virtual Constraints

Yoshihiro Sawamura * **Shingo Kojima** * **Masami Iwase** ** **Shoshiro Hatakeyama** **

* *Computers and Systems Engineering, Tokyo Denki University Ishizaka, Hatoyama-cho, Hiki-gun, Saitama, 350-0394, Japan e-mail: {sawamura, kojima}@hatalab.k.dendai.ac.jp*
** *Department of Robotics and Mechatronics, Computers and Systems Engineering, Tokyo Denki University 2-2 Nishiki-cho, Kanda, Tiyoda-ku, Tokyo, 101-8457, Japan e-mail: {iwase, sho}@fr.dendai.ac.jp, {iwase, sho}@k.dendai.ac.jp*

Abstract: In this study, we propose a nonlinear model-following control for underactuated systems. A model-following control design is based on the virtual constrains. The concept of virtual constraints control proposed by Shiriaev et. al. The model-following control is useful to realize the tracking system to a given reference model. In underactuated systems, a model-following control for fullactuated systems cannot be not utilized because there are limitations of coordinate transformation such as exact linearization. Thus this study deals with underactuated mechanical nonlinear systems. The design strategy is explained by taking a cart-pendulum as an example for this study. A cart-pendulum is controlled so that a virtual spring-mass-damper property is implemented to a pendulum. Some numerical simulations are performed to verify the effectiveness of the proposed method.

Keywords: virtual constraints, model-following control, projection method, underactuated nonlinear system

1. INTRODUCTION

Mechanical systems with many functions have been developed in recent years, and these systems work well in various areas. The manipulator is a typical example of such mechanical systems. Manipulators are utilized in so many factories as industry robots. One of important motions of a manipulator is to touch the external environment, and the impedance control is often used to avoid the destruction of the external environment by applying excess forces. The manipulator with the impedance control can behave as mass-spring-damper systems, and can realize soft touching. In order to improve the adaptivity and the flexibility, some control methods utilizing the virtual impedance between a manipulator and the external environment before contact have been proposed in Tsuji and Kaneko [1999], Tsuji et al. [2004]. If virtual characteristics are added to mechanical systems, various functions can be realized.

When desired virtual characteristics are implemented to some mechanical systems, the virtual characteristics are realized if the orbit of a target system follow the orbit of a reference system. In order to realize such control law, a model-following control is effective. Most of mechanical systems are nonlinear systems. Moreover, these systems may be underactuated systems. Particularly, in underactuated systems, a model-following control for fullactuated systems cannot be not utilized because there are limitations of coordinate transformation such as exact linearization.

Therefore, in this paper, we propose a nonlinear model-following control for underactuated system. In this study, we took a hint from the virtual constraint approach proposed by A.Shiriaev et al. [2005] , Shiriaev et al. [2006]. The virtual constraints control is a control method in which the orbit of a system should satisfy the given virtual constraints. A. Shiriaev et al proposed a stabilization control for underactuated systems. In the case of stabilizing a pendulum, an orbit of a pendulum is constrained to a limit cycle by using virtual constraints control. In other control methods which can attach virtual conditions to a target system, a control method which uses Controlled Lagrangian (A.M.Bloch et al. [2003], M.Kinoshita et al. [2006]) is proposed.

In our study, a virtual system can be considered as a reference model. A real system is a target system which is controlled. To track the output of the real system to the output of the virtual system , we introduce virtual holonomic constraints which denote the output of a real system and the output of a virtual system are coincident. These constraints are realized by designing control inputs which make the output of a real system and the output of virtual system coincident. An augmented system such as a model-following controller proposed by K.Komiya and K.Furuta [1982] is utilized in designing control inputs.

The model-following controller requires both model of the real system and the reference virtual system, and then the Projection Method is useful to derive these models. The Projection Method is a powerful modeling method, and we can find many works on the Projection Method in the literature: Blajer and Arczewski.K [1996], Blajer [2001], Ohata et al. [2004], Blajer [1992], Blajer [1995], H.Ohsaki et al. [2007]. The Projection Method takes the physical constraint conditions among sub-systems composing the entire system into account explicitly. The united model of the sub-systems can be derived by some algebraic calculations based on the described constraint conditions. Thus, the Projection Method permits us introducing virtual characteristics such as virtual forces easily.

This paper is organized as follows. In the preliminary section, the Projection Method is summaries briefly. The next section outlines virtual constraints control proposed by A.Shiriaev briefly. After that, a nonlinear model following control for underactuated systems is shown in the section 3. To verify the effectiveness of the proposed method, some numerical simulation are performed. Finally, we give some remarks in the conclusion.

2. PRELIMINARY

This study focuses on a type of the model-following control for nonlinear underactuated mechanical systems. The model-following controller requires both model of the real system and the reference virtual system. This section outlines The Projection Method.

The Projection Method is a powerful modeling method, and many studies on the Projection Method can be found nowadays (K.Watanabe et al. [2007], S.Terashima et al. [2003]). The Projection Method has the advantage that it can derive both the equations of motion and internal constrains of systems. This advantage allows us deriving easily a model which has virtual characteristics such as virtual forces.

The outline of the modeling procedure based on the Projection Method is summarized as follows:

(1) Set the coordinate system.
(2) Define q as a generalized coordinate, and v_q as a generalized velocity which sometimes includes quasi-velocities.
(3) Write down equations of motion of each component composing some system. Practically each component has a connection among other components to compose the entire system, however, the equation of motion of unconnected (free) component is considered in this step.
(4) Define M_q as the generalized mass matrix, and h as the generalized force according to the equations of motion in the previous step.
(5) Derive the holonomic and non-holonomic constraints showing the relation among components as $C_q v_q = 0$ using the constraint matrix C_q.
(6) Decompose the generalized velocity vector v_q to the tangent velocity v, which decides the degree of freedom of the constrained entire system, and the left velocity vector.
(7) Define D_q as the orthogonal complement matrix to C_q fulfilling $v_q = D_q\dot{v}$ and $C_q D_q = 0$.
(8) Derive the equations of motion of the constrained entire system by some algebraic calculation using D_q.

3. VIRTUAL CONSTRAINTS CONTROL

In this study, we utilize virtual constraints control proposed by A.Shiriaev et al. We explain a control method for realizing virtual holonomic constraints to a system.

A motion equation of general mechanical systems is represented by (1).

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = u \tag{1}$$

where the dimension of generalized coordinates q is n. $M(q), C(q,\dot{q}), G(q)$ are appropriate dimensions. And the dimension of inputs u is $n-1$ which is fewer than the degree of freedom of a system by one.

Then, defining new independent variable θ, (2) defines virtual holonomic constraints which is worked to the system (1).

$$q_1 = \psi_1(\theta), \cdots, q_n = \psi_n(\theta). \tag{2}$$

By defining new variables $y_1, \cdots, y_n$, the errors between virtual holonomic constraints and generalized coordinates are defined as

$$y_1 = q_1 - \psi_1(\theta),\ \cdots, y_n = q_n - \psi_n(\theta). \tag{3}$$

Then, introduce new coordinates y as follows:

$$y = [y_1, \cdots, y_{n-1}, \theta]^T. \tag{4}$$

q_n can be represented by

$$q_n = \psi_n(\theta) + h(y_1, \cdots, y_{n-1}, \theta), \tag{5}$$

so one and two differentiation of generalized coordinates q can be represented by one and two differentiation of new coordinates y. $\dot{q}$ and $\ddot{q}$ are shown in equations (6)(7) respectively.

$$\dot{q} = L\dot{y} \tag{6}$$

$$\ddot{q} = L\ddot{y} + N, \tag{7}$$

where

$$L = \begin{bmatrix} I_{n-1} & 0_{(n-1)\times 1} \\ \multicolumn{2}{c}{grad\ h} \end{bmatrix} + [0_{n\times(n-1)}, \Psi'] \tag{8}$$

$$N = L\dot{y} \tag{9}$$

$$\Psi' = [\psi_1', \psi_2', \cdots, \psi_n']. \tag{10}$$

A motion equation shown in (1) can transform to a motion equation (11) by using the relations (6)(7). ((q) and so on are abbreviated)

$$ML\ddot{y} + MN + CL\dot{y} + G = u \ \Big|_{\substack{q_1 = y_1 + \psi_1 \\ \vdots \\ q_n = y_n + \psi_n}} . \tag{11}$$

By solving (11) for $\ddot{\bar{y}} = [\ddot{y}_1, \cdots, \ddot{y}_{n-1}]^T$, (13) is obtained.

$$\begin{aligned} \ddot{\bar{y}} &= [I_{n-1}, 0]L^{-1}M^{-1}u \\ &\quad - [I_{n-1}, 0]L^{-1}M^{-1}(MN + CL\dot{y} + G) \quad (12) \\ &= Ku - R. \quad (13) \end{aligned}$$

Thus, when a control input u is defined following equation :

$$u = K^{-1}(v_y - R), \tag{14}$$

equation (13) can be written in partly linear form (15).

$$\ddot{\tilde{y}} = v_y. \tag{15}$$

Therefore, virtual constraints are realized to the system by using the state feedback which $[\tilde{y}, \dot{\tilde{y}}]$ converges to 0.

4. VIRTUAL CONSTRAINT BASED MODEL-FOLLOWING CONTROL FOR NONLINEAR SYSTEMS

A model-following controller based on the concept of virtual constraints is shown in this section. the model-following control is a control method which makes outputs of the real/reference system coincident. The reference system is considered as a virtual system in this paper. The virtual system has virtual characteristics which should be implemented to the real system. the controller shown in this paper requires that all states of the real system are known. The model-following control is realized by designing control inputs which make the outputs of the real/reference system coincident. An augmented system which consists of the real system and the virtual system is utilized to design the control inputs which make virtual constraints which denote the outputs of the real/reference system are coincident.

4.1 Augmented system which consists of the real system and the virtual system

We utilize the state dependent riccati equations proposed by J.R.Cloutier [1997], Cloutier and Cockburn [2001] for designing the control inputs. The state dependent linear representation is required to represent the real system and the virtual system.

A state dependent linear representation of the real system which is written by (16) is (17).

$$\begin{aligned} \dot{x}_a &= f(x_a) + g(x_a)u \\ y_a &= h(x_a) \end{aligned} \tag{16}$$

$$\Leftrightarrow$$

$$\begin{aligned} \dot{x}_a &= A(x_a)x_a + B(x_a)u_a \\ y_a &= C(x_a)x_a. \end{aligned} \tag{17}$$

Similarly, a state dependent linear representation of the virtual system is (18).

$$\begin{aligned} \dot{x}_m &= A(x_m)x_m + B(x_m)u_m \\ y_m &= C(x_m)x_m, \end{aligned} \tag{18}$$

where $x_a,\ x_m,\ u_a,\ u_m,\ y_a,\ y_m$ are the state , the input and the output for the real system and the virtual system respectively.

Then, an augmented system which consists of the real system and the virtual system is defined as (19).

$$\begin{aligned} \begin{bmatrix} \dot{x}_a \\ \dot{x}_m \end{bmatrix} &= \begin{bmatrix} A_a(x_a) & \\ & A_m(x_m) \end{bmatrix} \begin{bmatrix} x_a \\ x_m \end{bmatrix} + \\ & \begin{bmatrix} B_a(x_a) & \\ & B_m(x_m) \end{bmatrix} \begin{bmatrix} u_a \\ u_m \end{bmatrix} \\ \Leftrightarrow \dot{x} &= A(x)x + B(x)u. \end{aligned} \tag{19}$$

4.2 The virtual constraints which coincide the output of the real system and the virtual system

The augmented system written by (19) consists of the real system and the virtual system. Thus, if the real system is constrained virtually so that the output of the real system and the output of the virtual system are coincident, the output of the real system follows to the output of the virtual system. The virtual constraint can be obtained as (20) by the output equations of the real system and the virtual system.

$$y_a = y_m \ \Leftrightarrow\ C_a(x_a)x_a = C_m(x_m)x_m \tag{20}$$

4.3 Transform to error coordinates

Error coordinates ϵ are defined as (21).

$$\epsilon = C_a(x_a)x_a - C_m(x_m)x_m \tag{21}$$

If $C_a(x_a)$ is the column full rank matrix, the state x of the augmented system can be transformed to a new state $z = [\epsilon\ , x_m]^T$ which includes the virtual system state x_m and error coordinates ϵ. This transform can be represented as follows using pseudo inverse matrix of $C_a(x_a)$.

$$\begin{bmatrix} x_a \\ x_m \end{bmatrix} = \begin{bmatrix} C_a(x_a)^\dagger & C_a(x_a)^\dagger C_m(x_m) \\ O & I \end{bmatrix} \begin{bmatrix} \epsilon \\ x_m \end{bmatrix} \tag{22}$$

$$\Leftrightarrow x = L(x)z\ . \tag{23}$$

In a similar way , a transform equation between $\dot{x}$ and $\dot{z}$ is represented as follows.

$$\begin{aligned} \begin{bmatrix} \dot{x}_a \\ \dot{x}_m \end{bmatrix} &= \begin{bmatrix} C_a(x_a)^\dagger & C_a(x_a)^\dagger C_m(x_m) \\ O & I \end{bmatrix} \begin{bmatrix} \dot{\epsilon} \\ \dot{x}_m \end{bmatrix} + \\ & \begin{bmatrix} \dfrac{d\ C_a(x_a)^\dagger}{dt}\epsilon + \dfrac{d\ C_a(x_a)^\dagger}{dt}C_m(x_m)x_m + C_a(x_a)^\dagger \dfrac{d\ C_m(x_m)}{dt}x_m \\ O \end{bmatrix} \end{aligned} \tag{24}$$

$$\Leftrightarrow \dot{x} = L(x)\dot{z} + N(x, \dot{x})\ . \tag{25}$$

If $C_a(x_a)$ is not column full rank matrix, the state x should be divided as observable state x_{a2} and non-observable state x_{a1}. A state transformation equation can be obtaind by using the divided state.

The output of the real system can be reprensed as follows.

$$y_a = \begin{bmatrix} O & \tilde{C}_a \end{bmatrix} \begin{bmatrix} x_{a1} \\ x_{a2} \end{bmatrix}\ . \tag{26}$$

When the new state z is defined as $z = [x_{a1}\ , \epsilon\ , x_m]^T$, the transform equation between x and z is represented as follows.

$$\begin{bmatrix} x_{a1} \\ x_{a2} \\ x_m \end{bmatrix} = \begin{bmatrix} I & O & O \\ O & \tilde{C}_a(x_a)^\dagger & \tilde{C}_a(x_a)^\dagger C_m(x_m) \\ O & O & I \end{bmatrix} \begin{bmatrix} x_{a1} \\ \epsilon \\ x_m \end{bmatrix}$$

$$\Leftrightarrow x = L(x)z\ . \tag{27}$$

In a similar way , a transform equation between $\dot{x}$ and $\dot{z}$ is represented as follows.

$$\begin{bmatrix} \dot{x}_{a1} \\ \dot{x}_{a2} \\ \dot{x}_m \end{bmatrix} = \begin{bmatrix} I & O & O \\ O & \tilde{C}_a(x_a)^\dagger & \tilde{C}_a(x_a)^\dagger C_m(x_m) \\ O & O & I \end{bmatrix} \begin{bmatrix} \dot{x}_{a1} \\ \dot{\epsilon} \\ \dot{x}_m \end{bmatrix} + \begin{bmatrix} O \\ \frac{d\, C_a(x_a)^\dagger}{dt}\epsilon + \frac{d\, C_a(x_a)^\dagger}{dt} C_m(x_m)x_m + C_a(x_a)^\dagger \frac{d\, C_m(x_m)}{dt} x_m \\ O \end{bmatrix} \tag{28}$$

$$\Leftrightarrow \dot{x} = L(x)\dot{z} + N(x, \dot{x}) \ . \tag{29}$$

4.4 Control design

The state x of the augmented system represented by (19) can be transformed to the state z which includes error coordinates by using the state transformation equations shown in former subsection. Thus, following equation is obtained.

$$\dot{x} = A(x)x + B(x)u \tag{30}$$
$$\Leftrightarrow L(x)\dot{z} + N(x, \dot{x}) = A(x)L(x)z + B(x)u \tag{31}$$
$$\Leftrightarrow \dot{z} = L(x)^{-1}A(x)L(x)z + L(x)^{-1}B(x)u - L(x)^{-1}N(x, \dot{x}) \tag{32}$$
$$\Leftrightarrow \dot{z} = \bar{A}(x)z + \bar{B}(x)u - \bar{N}(x, \dot{x}) \ , \tag{33}$$

where

$$\bar{A}(x) = L(x)^{-1}A(x)L(x) \tag{34}$$
$$\bar{B}(x) = L(x)^{-1}B(x) \tag{35}$$
$$\bar{N}(x, \dot{x}) = L(x)^{-1}N(x, \dot{x}) \ . \tag{36}$$

Then, (33) can be rewritten as

$$\dot{z} = \bar{A}(x)z + \bar{B}(x)u - \bar{N}(x, \dot{x}) \tag{37}$$

$$\Leftrightarrow \begin{bmatrix} \dot{x}_{a1} \\ \dot{\epsilon} \\ \hline \dot{x}_m \end{bmatrix} = \begin{bmatrix} \bar{A}_{11}(x) & \bar{A}_{12}(x) \\ O & A_m(x) \end{bmatrix} \begin{bmatrix} x_{a1} \\ \epsilon \\ \hline x_m \end{bmatrix} + \begin{bmatrix} \bar{B}_{11}(x) & \bar{B}_{12}(x) \\ O & B_m(x) \end{bmatrix} \begin{bmatrix} u_a \\ u_m \end{bmatrix} - \begin{bmatrix} \bar{N}_1(x, \dot{x}) \\ \bar{N}_2(x, \dot{x}) \end{bmatrix} \tag{38}$$

$$= \begin{bmatrix} \bar{A}_{11}(x) & \bar{A}_{12}(x) \\ O & A_m(x) \end{bmatrix} \begin{bmatrix} x_{a1} \\ \epsilon \\ \hline x_m \end{bmatrix} + \begin{bmatrix} \bar{B}_{11}(x) \\ O \end{bmatrix} u_a + \begin{bmatrix} \bar{B}_{12}(x) \\ B_m(x) \end{bmatrix} u_m - \begin{bmatrix} \bar{N}_1(x, \dot{x}) \\ \bar{N}_2(x, \dot{x}) \end{bmatrix} \ . \tag{39}$$

If $\bar{B}_{11}(x)$ is row full rank matrix, the input u_a for the real system can be defined as

$$u_a = \bar{u}_a - \begin{bmatrix} \bar{B}_{11}(x) \\ O \end{bmatrix}^\dagger \left(\begin{bmatrix} \bar{B}_{12}(x) \\ B_m(x) \end{bmatrix} u_m - \begin{bmatrix} \bar{N}_1(x, \dot{x}) \\ \bar{N}_2(x, \dot{x}) \end{bmatrix} \right) \ . \tag{40}$$

If $\bar{B}_{11}(x)$ is not row full rank matrix, a following term is treated as a disturbance.

$$\begin{bmatrix} \bar{B}_{12}(x) \\ B_m(x) \end{bmatrix} u_m - \begin{bmatrix} \bar{N}_1(x, \dot{x}) \\ \bar{N}_2(x, \dot{x}) \end{bmatrix} \tag{41}$$

Then, the input u_a for the real system can be defined as

$$u_a = \bar{u}_a \ . \tag{42}$$

Thus, (39) can be rewritten as

$$\begin{bmatrix} \dot{x}_{a1} \\ \dot{\epsilon} \\ \hline \dot{x}_m \end{bmatrix} = \begin{bmatrix} \bar{A}_{11}(x) & \bar{A}_{12}(x) \\ O & A_m(x) \end{bmatrix} \begin{bmatrix} x_{a1} \\ \epsilon \\ \hline x_m \end{bmatrix} + \begin{bmatrix} \bar{B}_{11}(x) \\ O \end{bmatrix} \bar{u}_a$$
$$\Leftrightarrow \dot{z} = A_{mf}(x)z + B_{mf}(x)\bar{u}_a \ . \tag{43}$$

The error coordinates ϵ have to converge to 0 so that the virtual constraints represented by (20) are effected to the augmented system (43). Thus, in order to derive the input for the real system, a following evaluation function is introduced.

$$J = \int_0^\infty \left(z^T Q z + \bar{u}_a^T R \bar{u}_a \right) dt \tag{44}$$

R is a weight matrix for the input. Q is a weight matrix for the state and can be represented as

$$Q = \begin{bmatrix} Q_1 & O \\ O & O \end{bmatrix} \ . \tag{45}$$

The input which makes (44) minimum is given as following

$$\bar{u}_a = -F_1[x_{a1}^T, \epsilon^T]^T - F_2 x_m \ , \tag{46}$$

where

$$F_1 = R^{-1}\bar{B}_{11}(x)P_11(x) \tag{47}$$
$$F_2 = R^{-1}\bar{B}_{11}(x)P_12(x). \tag{48}$$

And $P_{11}(x)$, $P_{12}(x)$ are positive definite symmetric matrix and satisfy following equations.

$$0 = \bar{A}_{11}(x)^T P_{11}(x) + P_{11}(x)\bar{A}_{11}(x) + Q - P_{11}(x)\bar{B}_{11}(x)R^{-1}\bar{B}_{11}(x)^T P_{11}(x) \tag{49}$$
$$0 = \bar{A}_{11}(x)^T P_{12} + P_{11}\bar{A}_{12}(x) + P_{12}A_m(x) - P_{11}(x)\bar{B}_{11}(x)R^{-1}\bar{B}_{11}(x)^T P_{12}(x) \tag{50}$$

Therefore, the input of real system can be derived by (40) and (46).

4.5 Zero dynamics

If the control input for the real system makes the virtual constraints (20) invariant, the state x_{a1} which is not transformed to error coordinates has zero dynamics.

If x_{a1} is represned as

$$\dot{x}_{a1} = \alpha(x_{a1},\ x_{a2}) + \beta(x_{a1},\ x_{a2})u_a \ . \tag{51}$$

then the zero dynamics can be represented as

$$\dot{x}_{a1} = \alpha(x_{a1},\ x_{a2}) \ \Big|_{x_{a2} = \tilde{C}_a(x_a)^\dagger C_m(x_m)x_m} \tag{52}$$

Table 1. Parameters

Items	Value
Mass of cart m_c	0.5[kg]
Mass of pendulum m_p	0.01[kg]
Length of pendulum l	0.2[m]
Inertia of pendulum j	0.005[kg · m^2]
Viscous friction of pendulum c_p	0.0003
Viscous friction of cart c_c	0.0001
Mass of spring-mass-damper system m_v	1.0[kg]
Spring modulus k	0.1
Damper modulus d	0.01

5. SIMULATION AND VALIDATION

A Simulation is presented to verify the effectiveness of the proposed method. In a simulation shown in this section, the real system is the cart-pendulum system which is a underactuated nonlinear system. And the virtual system is the spring-mass-damper system. In this simulation, we design a controller so that the angle and angular velocity of the cart-pendulum system are coincident with the output of the virtual system. The parameters which are utilized in the simulation are shown in table.5.

5.1 The real system and the virtual system used in simulation

The state dependent linear representation of the cart-pendulum system is shown in following.

$$\dot{x}_a = \begin{bmatrix} O & I \\ -M^{-1}G & -M^{-1}C \end{bmatrix} x_a + \begin{bmatrix} O \\ M^{-1}[0\ 1]^T \end{bmatrix} u_a \tag{53}$$

$$y_a = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} x_a\ ,\ x_a = [\theta\ l_x\ \dot{\theta}\ \dot{l}_x]^T\ , \tag{54}$$

where

$$M(\theta) = \begin{bmatrix} j + m_p l^2 & m_p l \cos\theta \\ m_p l \cos\theta & m_c + m_p \end{bmatrix} \tag{55}$$

$$C(\theta,\ \dot{\theta}) = \begin{bmatrix} c_p & 0 \\ m_p l \sin\theta\dot{\theta} & c_c \end{bmatrix} \tag{56}$$

$$G(\theta) = \begin{bmatrix} m_p g l \sin\theta/\theta & 0 \\ 0 & 0 \end{bmatrix}\ . \tag{57}$$

The state space representation of the virtual system is shown in following.

$$\dot{x}_m = \begin{bmatrix} 0 & 1 \\ -k/m_v & -d/m_v \end{bmatrix} x_m + \begin{bmatrix} 0 \\ 1/m_v \end{bmatrix} u_m$$
$$\Leftrightarrow \dot{x}_m = A_m x_m + B_m u_m$$
$$y_m = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_m\ ,\ x_m = [x_1\ \dot{x}_1]^T\ . \tag{58}$$

Then, a virtual constraint which makes the output of the real system and the output of the virtual system coincident is represented as

$$\epsilon = [O,\ \tilde{C}_a] \begin{bmatrix} x_{a1} \\ x_{a2} \end{bmatrix} - C_m x_m \tag{59}$$
$$= \tilde{C}_a x_{a2} - C_m x_m\ , \tag{60}$$

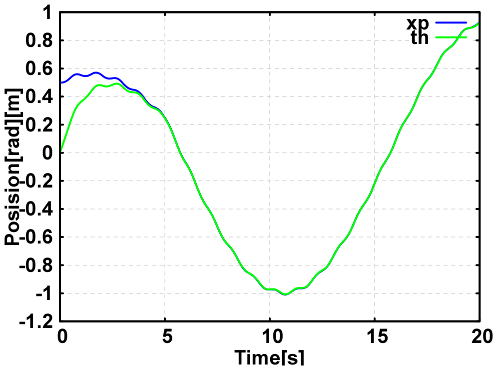

Fig. 1. An angle of the pendulum and a position of spring-mass-damper system. (blue line: the virtual system, green line: the real system)

where $x_{a1} = [l_x,\ \dot{l}_x]^T$, $x_{a2} = [\theta,\ \dot{\theta}]^T$ and

$$\tilde{C}_a = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\ . \tag{61}$$

In order to utilize the state transformation equation which is shown in (27) and (29), the state of the real system is required to be transformed by using the following state transformation matrix. The state transformed real system is utilized for designing a controller.

$$T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{62}$$

5.2 Simulation result

In this simulation, the initial condition of the real system is $x_a(0) = [0,\ 0,\ 0,\ 0]^T$, the initial condition of the virtual system is $x_m(0) = [0.5,\ 0]^T$. The control interval is 20[msec], And we utilize the following input for the virtual system.

$$u_m = \begin{cases} 0.5\sin(2\pi t) & ,\ t < 5.0 \\ -0.5\sin(2\pi t) & ,\ t >= 5.0 \end{cases}\ . \tag{63}$$

The weight matrices which used in (44) are

$$Q_1 = \mathrm{diag}(150.0,\ 0.0,\ 140.0,\ 0.0) \tag{64}$$
$$R = 0.005\ . \tag{65}$$

The simulation results are shown in Fig.1.2.3 In Fig.1 and 2, the blue line is the trajectory of the virtual system. The green line is the trajectory of the real model. Fig.1 and Fig.2 show that the angle and angular velocity of the pendulum can track the position and velocity of the virtual model almost perfectly.

5.3 Zero dynamics of a cart

If the feedback input makes the virtual constraints invariant, following relations are obtained by (60).

$$\theta = x_1\ ,\ \dot{\theta} = \dot{x}_1\ ,\ \ddot{\theta} = \ddot{x}_1\ .$$

Thus, zero dynamics of a cart is represented as following equation when the virtual constraints are invariant.

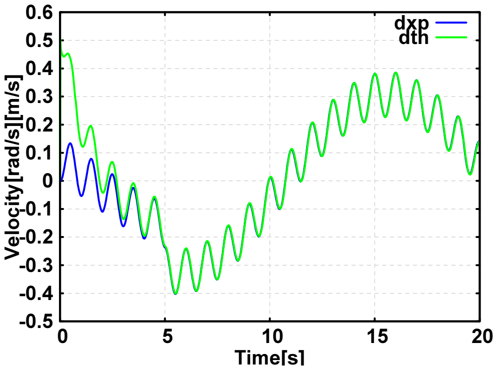

Fig. 2. An angular velocity of the pendulum and a velocity of spring-mass-damper system. (blue line: the virtual system, green line: the real system)

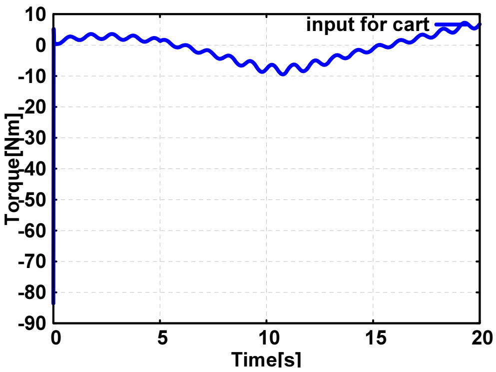

Fig. 3. An input for the cart.

$$(m_p l \cos x_1)\,\ddot{x}_1 + (m_c + m_p)\,\ddot{l}_x + (m_p l \sin x_1)\,\dot{x_1}^2 + c_c\dot{l}_x = 0 \ . \qquad (66)$$

Then, if the position and velocity of the virtual system which is the spring-mass-damper system are stabilized to 0 , zero dynamics of a cart is represented as following

$$(m_c + m_p)\,\ddot{l}_x + c_c\dot{l}_x = 0 \ , \qquad (67)$$

where m_c, m_p, c_c are positive constant. Thus, (67) is stable.

If a motion of the virtual system is around an origin, the time response of the spring-mass-damper system is represented as

$$x_1(t) = \int_0^t g(t-\tau)u_m(\tau)d\tau \ , \qquad (68)$$

$$g(t) = \mathcal{L}^{-1}\{G(s)\} \ ,$$

$$G(s) = \frac{1}{(s-\alpha)(s-\beta)} \ , \ \ \Re(\alpha) < 0, \ \ \Re(\beta) < 0$$

the virtual system is a stable. Then, the time response of l_x can be obtained as following equation by linearizing (66) around $x_1 \simeq 0$.

$$l_x(t) = \int_0^t \left(\frac{a_3}{a_1} + \left(a_2 - \frac{a_3}{a_1}\right)e^{-a_1(t-\tau)}\right)x_1(\tau)d\tau (69)$$

a_1, a_2, , a_3 are positive constant.

Thus, l_x is represented by the convolution integral of multiplication of non-divergent term and stable term. l_x is stable.

6. CONCLUSION

This paper presents a model-following control of underactuated nonlinear systems based on virtual constraints. To realize the virtual characteristics, the control method lets the output of the real system follow the output of the virtual reference. The Projection method can be utilized for the derivation of models of the systems which has which has virtual characteristics. These control system design has been explained by using a cart-pendulum system as an example of this study. Some numerical simulations have been performed to confirm that the real system tracks the virtual system , and the effectiveness has been verified.

REFERENCES

A.M.Bloch, J.Baillieul, P.Crouch, and J.E.Marsden. *Nonholonomic Mechanics and Control*. Springer, 2003.

A.Shiriaev, J.W.Perram, and C.Canudas de Wit. Constructive tool for orbital stabilization of underactuated nonlinear systems: virtual constraintsapproach. *IEEE Transactionson Automatic Control*, 50(8):1164–1176, 2005.

W. Blajer. A projection method approach to constrained dynamic analysis. J. Appl. Mech. 59, pages 51–64, 1992.

W. Blajer. Projective formulation of lagrange's and boltzmann-hamel equations for multibody systems. *ZAMM 75 SI*, pages 107–108, 1995.

W. Blajer. A geometrical interpretation and uniform matrix formulationof multibody system dynamics. *ZAMM 81*, 4:247–259, 2001.

W. Blajer and Arczewski.K. A unified approach to the modeling ofholonomic and nonholonomic mechanical systems. volume 2 of *Mathematical Modeling of Systems*, pages 157–174, 1996.

J. R. Cloutier and J.C. Cockburn. The state-dependent nonlinear regulator with state constraints. American Control Conference, pages 390–395, 2001.

H.Ohsaki, M.Iwase, and S.Hatakeyama. A consideration of nonlinear system modeling using the projection method. SICE Annual Conference 2007, 9 2007.

J.R.Cloutier. State-dependent riccati equation techniques: Anoverview. American Control Conference, pages 932–936, 1997.

K.Komiya and K.Furuta. Design of model-following servo controller. *IEEE Transactions on Auto-matic Control*, 27(3):725–727, 1982.

K.Watanabe, M.Iwase, S.Hatakeyama, and T.Maruyama. Control strategy for a snake-like robot based on constraint force and its validation. Number 155 in 2007 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2007. in USB.

M.Kinoshita, M.Iwase, and S.Hatakeyama. Stabilization of inverted pendulum on a cart with slope conditions by using controlled lagrangian. volume 49 of *49th Jido Seigyo Rengo Koenkai*, pages SA9–2–2, 2006. in Japanese.

A. Ohata, A. Sugiki, Y. Ohta, S. Suzuki, and K. Fnruta. Engine modeling based on projection method and conservation laws. 2004 IEEE International Conference on Control Applications, 9 2004.

A. Shiriaev, L.Freidovich, A.Robertsson, and R.Johansson. Virtual-constraints-based design of stable oscillations of furuta pendulum: Theory and experiments. CDC '06, pages 6144–6149, 2006.

S.Terashima, M.Iwase, K.Furuta, S.Suzuki, and S.Hatakeyama. A design of servo controller for nonlinear systems using state dependent riccati equation. volume 4 of *Decision and Control, 2003. Proceedings. 42nd IEEE Conference*, pages 3864–3869, 2003.

T. Tsuji and M. Kaneko. Non-contact impedance control for redundant manipulator. *IEEE Trans. Syst.*, 29:184–193, 3 1999.

T. Tsuji, M. Terauchi, and Y.Tanaka. Online learning of virtual impedance parameters in non-contact impedance control using neural networks. *IEEE Trans. Syst*, 34:2112–2118, 10 2004.