The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Basis-Motion Torque Composition Approach: Generation of Feedforward Inputs for Control of Multi-Joint Robots

Masahiro Sekimoto, Sadao Kawamura, Tomoya Ishitsubo, Shinsuke Akizuki, and Masayuki Mizuno

Abstract— **This paper focuses on generation of feedforward torque for precise joint-trajectory tracking of a multi-joint robot arm with serially connected links. A proposed method called basis-motion torque composition, based on four arithmetical operations of time-series torque data for several motions, allows to generate feedforward torque for a motion whose final posture and time profile are specified. The torque-generation algorism is presented, and the experimental results by a two-joint robot arm are illustrated. The experimental results demonstrate that the tracking errors of angular velocities by the basis-motion torque composition tend to be smaller than those by the computed torque method.**

I. INTRODUCTION

During fast movements of a multi-joint robot with serially connected links, an inertia-induced force (i.e., inertia, centrifugal, and Coriolis forces) is dominant. Hence, the force often incurs the difficulty of dynamic robot control. For precise trajectory tracking of robot under such a condition, feedforward control is efficient. The computed torque method is widely known as one of the feedforward control approaches in robotics. The method is significantly efficient if a dynamics structure of a robot including drive systems and all of its dynamics parameters are known. However, for a robot with a complicated structure like a humanoid robot, it is increasingly difficult to accurately evaluate all of the dynamics parameters in practice, though the estimation schemes of dynamics parameters have been suggested [1]–[4].

On the other hand, the iterative learning control (ILC) is also widely known as another approach to achieve precise trajectory tracking. A desired task is accomplished after repetitions of trials due to feedforward inputs built by an iterative learning update law. An early use of ILC for mechanical systems can be found in a U.S. patent [5] filed in 1967 as well as the Japanese journal paper published in 1978 [6]. Later, Arimoto *et al.* [7] have formulated the ILC in a set of axioms and given an explicit sufficient condition for convergence of learning, then it has been widely investigated in control of repetitive tasks for not only robots but also various kinds of mechatronics systems [8], [9]. The original ILC has been recently extended to the task space ILC for endpoint trajectory tracking of a redundant robot [10], [11]. Since an iterative learning update law is generally constructed from only kinematic information and state-variable information, the ILC realizes a desired motion without estimation of individual dynamics parameters. However, what another learning process is needed according to change of task, has been often criticized in comparison to the computed torque method.

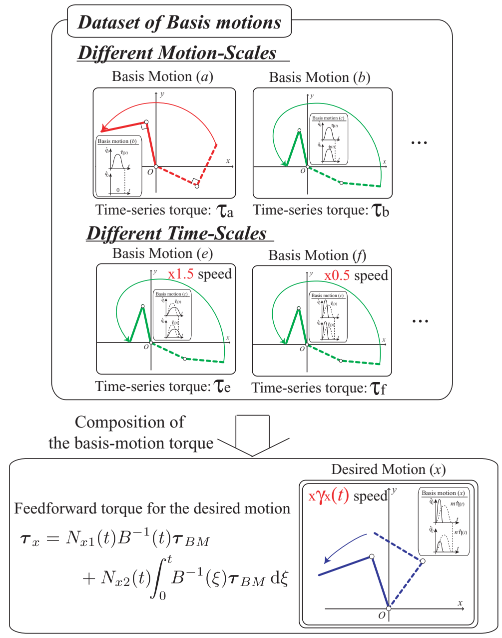


Fig. 1. An Overview of basis-motion torque composition

Against the criticism, Kawamura *et al.* have suggested the scheme named *time-scale transformation (TST)* to generate a specified motion without any additional learning processes for change of a desired motion nor any priori knowledge of dynamics parameters [12], [13]. The TST allows to generate feedforward torque for a motion with a specified velocity profile by performing arithmetic operations of time-series torque data obtained preliminarily by ILC, though the class of generable torque of TST is restricted to motions determined by extending or shortening a time duration of reference motion. On the other hand, Sekimoto *et al.* have recently suggested the scheme named *motion-scale transformation (MST)* to generate a motion to a specified posture on the basis of arithmetic operations of time-series torque data [14]. The MST allows to generate desired feedforward toque by assuming a motion profile and a dynamics structure. Hence, the class of generable torque of MST is restricted to motions with the fixed movement time and the fixed motion profile.

This paper aims at extending the class of generable

This work was partially supported by MEXT KAKENHI (No.20033021): Grant-in-Aid for Scientific Research on Priority Areas "Mobiligence."

M. Sekimoto and S. Kawamura are with the Research Organization of Science and Engineering, Ritsumeikan University, 1-1-1 Nojihigashi, Kusatsu, Shiga 525-8577, Japan sekimoto@fc.ritsumei.ac.jp

S. Kawamura, S. Akizuki, T. Ishitsubo, and M. Mizuno are with the department of Robotics, Ritsumeikan University, 1-1-1 Nojihigashi, Kusatsu, Shiga 525-8577, Japan

978-1-4244-3804-4/09/$25.00 ©2009 IEEE

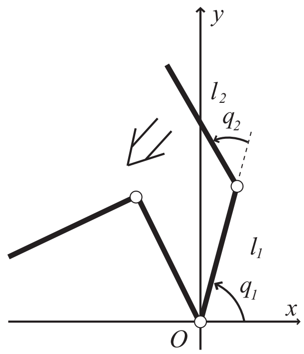


Fig. 2. Planar movement of a robot arm with two joints

motions without any additional learning processes for target change nor any priori knowledge of dynamics parameters, by combining the TST and the MST (see Fig.1). The method named *basis-motion torque composition* allows to generate feedforward torque for a motion whose final posture and time profile are specified. The torque-generation algorism is presented, and the experimental results by a two-joint robot arm are illustrated. The experimental results demonstrate that the tracking errors of angular velocities by the basis-motion torque composition tend to be smaller than those by the computed torque method.

II. PROBLEM STATEMENT

A. ASSUMPTIONS OF DYNAMICS

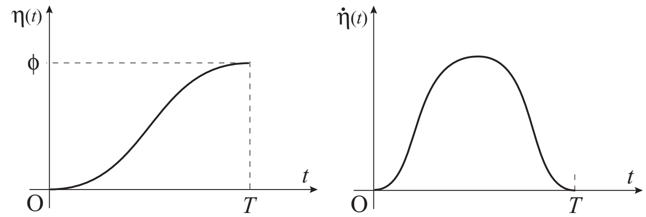
Our objective is to generate feedforward torque input for a multi-joint robot to track a desired joint trajectory $\boldsymbol{q}_r(t) \in \Re^2$ precisely. Let us consider planar movement of a robot arm with two joints as shown in Fig.2. Firstly, we assume that Lagrange's equation of motion of the robot arm including its drive systems can be described by

$$H(\boldsymbol{q})\ddot{\boldsymbol{q}}+\left\{\frac{1}{2}\dot{H}(\boldsymbol{q})+S(\boldsymbol{q},\dot{\boldsymbol{q}})\right\}\dot{\boldsymbol{q}}+B\dot{\boldsymbol{q}}+\boldsymbol{f}_c(\dot{\boldsymbol{q}})+\boldsymbol{\xi}(t)=\boldsymbol{u} \quad (1)$$

where $\boldsymbol{q} = (q_1, q_2)^{\mathrm{T}}$ denotes the vector of joint angles, $H(\boldsymbol{q}) \in \Re^{2\times 2}$ denotes the inertia matrix, $S(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}$ denotes the gyroscopic force term including centrifugal and Coriolis forces, $S(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \Re^{2\times 2}$ denotes the skew-symmetric matrix, $B\dot{\boldsymbol{q}} + \boldsymbol{f}_c(\dot{\boldsymbol{q}}) \in \Re^2$ denotes the joint-friction force, $B \in \Re^{2\times 2}$ denotes the positive definite and diagonal matrix, $\boldsymbol{u} \in \Re^2$ denotes the control input torque at joints [8]. In practice, there exists a poorly-reproducible effect caused by temperature-dependent change of joint-frictional properties during robot movements, though the effect is small unless a robot works under the high- or low-temperature condition. The element $\boldsymbol{\xi}(t)$ denotes all of the poorly-reproducible and time-dependent effects. However, a feedforward input cannot cope with the influences. In order to solve the problem, a hybrid controller composed of PD-feedback and feedforward inputs

$$\boldsymbol{u} = -K_v\Delta\dot{\boldsymbol{q}} - K_p\Delta\boldsymbol{q} + \boldsymbol{\tau} \quad (2)$$

is applied to the system. In eq.(2), K_v and K_p denote 2×2 positive definite and diagonal matrices for the feedback gains, $\Delta\boldsymbol{q} = \boldsymbol{q} - \boldsymbol{q}_r$, and $\boldsymbol{\tau} \in \Re^2$ is a feedforward torque



(a) $\eta(t)$ (b) $\dot{\eta}(t)$

Fig. 3. The profiles of motion primitive

input. The feedback gains are set with small values so that the feedforward input can be dominant. Secondly, we assume that the feedforward input acts so as to compensate the poor-reproducible effects (that is, so as to satisfy $\boldsymbol{\xi}(t) = -K_v\Delta\dot{\boldsymbol{q}}-K_p\Delta\boldsymbol{q}$) when the robot realizes the desired motion $\boldsymbol{q}_r(t)$ precisely. Then, the feedforward input $\boldsymbol{\tau}_r$ for realizing the desired motion satisfies the relation

$$\boldsymbol{\tau}_r = H(\boldsymbol{q}_r)\ddot{\boldsymbol{q}}_r+\left\{\frac{1}{2}\dot{H}(\boldsymbol{q}_r)+S(\boldsymbol{q}_r,\dot{\boldsymbol{q}}_r)\right\}\dot{\boldsymbol{q}}_r+B\dot{\boldsymbol{q}}_r+\boldsymbol{f}_c(\dot{\boldsymbol{q}}_r) \quad (3)$$

Hence, the objective is to generate the feedforward torque input.

Thirdly, we assume that a dynamics structure of robot is known but individual elements in dynamics are unknown. For instance, the inertia matrix $H(\boldsymbol{q})$, since every entry of $H(\boldsymbol{q})$ is a constant or a sinusoidal function of components of joint angle vector $\boldsymbol{q}$, is assumed as

$$H(\boldsymbol{q}) = \begin{bmatrix} a_{11}+2a\cos q_2 & a_{22}+a\cos q_2 \\ a_{22}+a\cos q_2 & a_{22} \end{bmatrix} \quad (4)$$

where a_{11}, a_{22}, and a are unknown dynamics parameters. Thus, we assume the desired feedforward input in eq.(3) as

$$\begin{bmatrix}\tau_{r1}\\ \tau_{r2}\end{bmatrix} = \begin{bmatrix} a_{11}+2ac_{r2} & a_{22}+ac_{r2} \\ a_{22}+ac_{r2} & a_{22}\end{bmatrix}\begin{bmatrix}\ddot{q}_{r1}\\ \ddot{q}_{r2}\end{bmatrix} + \left\{\frac{as_{r2}\dot{q}_{r2}}{2}\begin{bmatrix}-2 & -1\\ -1 & 0\end{bmatrix} + \frac{as_{r2}(2\dot{q}_{r1}+\dot{q}_{r2})}{2}\begin{bmatrix}0 & -1\\ 1 & 0\end{bmatrix}\right\}\begin{bmatrix}\dot{q}_{r1}\\ \dot{q}_{r2}\end{bmatrix} + \begin{bmatrix}d_1 & 0\\ 0 & d_2\end{bmatrix}\begin{bmatrix}\dot{q}_{r1}\\ \dot{q}_{r2}\end{bmatrix} + \begin{bmatrix}\rho_1\mathrm{sgn}(\dot{q}_{r1})\\ \rho_2\mathrm{sgn}(\dot{q}_{r2})\end{bmatrix} \quad (5)$$

where a_{11}, a_{22}, and a denote positive constants related to the inertia matrix, d_1, d_2, ρ_1, and ρ_2 denote positive constants related to the joint-friction force, $\boldsymbol{\tau}_r = (\tau_{r1}, \tau_{r2})^{\mathrm{T}}$, $s_{r2} = \sin q_{r2}$, $c_{r2} = \cos q_{r2}$, and $\mathrm{sgn}(\cdot)$ denotes a signum function. In the right hand side of eq.(5), the first term denotes the inertial force, the second term denotes the Coriolis force and the centrifugal force, and the third and fourth terms denote the joint-friction force corresponding to $B\dot{\boldsymbol{q}}_r + \boldsymbol{f}_c(\dot{\boldsymbol{q}}_r)$ in eq.(3).

B. ASSUMPTIONS OF MOTIONS

Motions treated in the paper are restricted to motions formulated on the basis of a smooth function with a finite time duration $t \in [0, T]$ defined by

$$\eta(t) = \phi\left[6\left(\frac{t}{T}\right)^5 - 15\left(\frac{t}{T}\right)^4 + 10\left(\frac{t}{T}\right)^3\right] \quad (6)$$

where ϕ denotes the magnitude of motion given as a positive constant and T denotes a terminal time of motion given as a positive constant. The position and velocity profiles of function are shown in Fig.3. For the sake of convenience, the function is called *a motion primitive*. Based on the motion primitive, a motion is described as

$$\begin{cases} q_{r1}(t) = m_r \eta(t) + q_{r01} \\ q_{r2}(t) = n_r \eta(t) + q_{r02} \end{cases} \tag{7}$$

where m_r and n_r denote constants for motion scales, and q_{r01} and q_{r02} denote constants for an initial pose of robot. Note that all of the motions treated in the paper follow the axiomatic characteristics of iterative learning (see the book [8]). The function in eq.(6) can be replaced with an arbitrary function of C^2 class.

C. TIME-SCALE TRANSFORMATION

The time-scale transformation allows to generate feedforward torque for a motion in specified speed by performing arithmetic operations of time-series torque data obtained by the ILC, though the class of generable torque is restricted to motions obtained by extending or shortening a time duration of reference motion [13].

Let us introduce another motion $\boldsymbol{q}_s \in \Re^2$ different from the motion in eq.(7) concerning motion speed. Then, since the kinematic joint-passes of both motions are same, it satisfies

$$\boldsymbol{q}_s(r_s(t)) = \boldsymbol{q}_r(t) \tag{8}$$

where $r_s(t)$ denotes a time-scale function related to the time t of the motion (r). Given a time duration $r_s(t) \in [0, T_s]$, the function $r_s(t)$ satisfies the conditions:

$$\begin{cases} \text{(i)} \quad r_s(0) = 0, \quad r_s(T) = T_s \\ \text{(ii)} \quad r_s(t) \in C^2 \quad \text{for} \;\; t \in [0, T] \\ \text{(iii)} \quad 0 < \frac{\mathrm{d}r_s(t)}{\mathrm{d}t} < \infty \quad \text{for} \;\; t \in [0, T] \end{cases} \tag{9}$$

Also, the velocity and acceleration of $\boldsymbol{q}_s(r_s(t))$ follow the relations

$$\begin{cases} \boldsymbol{q}_s'(r_s(t)) \left(= \dfrac{\mathrm{d}\boldsymbol{q}_s(r_s(t))}{\mathrm{d}r_s(t)} \right) = \alpha_s(t)\, \dot{\boldsymbol{q}}_r(t) \\ \boldsymbol{q}_s''(r_s(t)) \left(= \dfrac{\mathrm{d}^2\boldsymbol{q}_s(r_s(t))}{\mathrm{d}r_s(t)^2} \right) = \alpha_s^2(t)\, \ddot{\boldsymbol{q}}_r(t) - \alpha_s^3(t)\beta_s(t)\, \dot{\boldsymbol{q}}_r(t) \end{cases} \tag{10}$$

where $\alpha_s(t)$ and $\beta_s(t)$ are defined by

$$\alpha_s(t) = 1 \Big/ \frac{\mathrm{d}r_s(t)}{\mathrm{d}t}, \quad \beta_s(t) = \frac{\mathrm{d}^2 r_s(t)}{\mathrm{d}t^2} \tag{11}$$

Then, the feedforward torque for realizing the motion $\boldsymbol{q}_s(r_s(t))$ is written by

$$\boldsymbol{\tau}_s = H(\boldsymbol{q}_s)\boldsymbol{q}_s'' + \left\{ \frac{1}{2} H'(\boldsymbol{q}_s) + S(\boldsymbol{q}_s, \boldsymbol{q}_s') \right\} \boldsymbol{q}_s' + B\boldsymbol{q}_s' + \boldsymbol{f}_c(\boldsymbol{q}_s') \tag{12}$$

By substituting eqs.(8) and (10) into this equation, it can be rewritten as

$$\begin{aligned} \boldsymbol{\tau}_s = & \alpha_s^2 H(\boldsymbol{q}_r)\ddot{\boldsymbol{q}}_r + \alpha_s^2 \left\{ \frac{1}{2}\dot{H}(\boldsymbol{q}_r) + S(\boldsymbol{q}_r, \dot{\boldsymbol{q}}_r) \right\} \dot{\boldsymbol{q}}_r \\ & + \alpha_s B\dot{\boldsymbol{q}}_r + \boldsymbol{f}_c(\alpha_s \dot{\boldsymbol{q}}_r) - \alpha_s^3 \beta_s H(\boldsymbol{q}_r)\dot{\boldsymbol{q}}_r \end{aligned} \tag{13}$$

It is noteworthy that the feedforward torque for realizing the motion $\boldsymbol{q}_s(r_s(t))$ can be described based on the elements of dynamics for the motion $\boldsymbol{q}_r(t)$ and the time-scale function $r_s(t)$. Thus, by using the dynamic property in the differences of time scales, the time-scale transformation allows to generate feedforward torque for an arbitrary-speed motion from feedforward torque for realizing the four-time-scale motions (three linear ones and a nonlinear one).

D. PROBLEM

Under the assumptions described above, let us consider to generate feedforward torque for realizing a motion $\boldsymbol{q}_x(r_x(t)) = (q_{x1}(r_x(t)), q_{x2}(r_x(t)))^{\mathrm{T}}$ specified by

$$\begin{cases} q_{x1}(r_x(t)) = m_x \eta_x(r_x(t)) + q_{x01} \\ q_{x2}(r_x(t)) = n_x \eta_x(r_x(t)) + q_{x02} \end{cases} \tag{14}$$

over a time duration $r_x(t) \in [0, T_x]$ for given m_x, n_x, q_{x01}, q_{x02}, and $r_x(t)$. The time-scale function $r_x(t)$ is determined so as to satisfy the conditions in eq.(9). Also, the time-scale transformed motion primitive $\eta_x(r_x(t))$, based on the original motion primitive in eq.(6), is defined so as to satisfy

$$\begin{cases} \eta_x(r_x(t)) = \eta(t) \\ \eta_x'(r_x(t)) \left(= \dfrac{\mathrm{d}\eta_x(r_x(t))}{\mathrm{d}r_x(t)} \right) = \alpha_x(t)\, \dot{\eta}(t) \\ \eta_x''(r_x(t)) \left(= \dfrac{\mathrm{d}^2\eta_x(r_x(t))}{\mathrm{d}r_x(t)^2} \right) = \alpha_x^2(t)\, \ddot{\eta}(t) - \alpha_x^3(t)\beta_x(t)\, \dot{\eta}(t) \end{cases} \tag{15}$$

where $\alpha_x(t)$ and $\beta_x(t)$ are defined by

$$\alpha_x(t) = 1 \Big/ \frac{\mathrm{d}r_x(t)}{\mathrm{d}t}, \quad \beta_x(t) = \frac{\mathrm{d}^2 r_x(t)}{\mathrm{d}t^2} \tag{16}$$

The desired feedforward torque cannot be directly derived from eq.(5) due to unknown dynamics parameters. Even in the case, the ILC allows to simultaneously acquire the desired motion and the desired feedforward torque after repetitions. However, an additional learning process is required to realize another motion.

As shown in eq.(5), the time-series torque data obtained by the ILC correspond to the system dynamics, though they do not directly correspond to the elements of dynamics. Hence, by referring to the relations between the torque data obtained by the ILC and their dynamics properties, the desired feedforward torque should be available without any additional learning processes even in the case of change of a desired motion. Now, we suggest the following proposition.

Proposition — In the case of planar motions of a two-joint robot arm under the assumptions described above, for given ϕ and T, a motion primitive is defined by eq.(6), and four motions are chosen adequately based on the motion primitive. Feedforward torque for realizing the motions are obtained by the ILC, and four pairs of joint trajectories and time-series toque data $((\boldsymbol{q}_a(t), \boldsymbol{\tau}_a(t)), \ldots, (\boldsymbol{q}_d(t), \boldsymbol{\tau}_d(t)))$ are prepared. Then, for a desired motion in eq.(14) specified by given m_x, n_x, q_{x01}, q_{x02}, and $r_x(t)$, the corresponding feedforward torque can be obtained from arithmetic operations of the four dataset pairs (see Fig.1).

For the sake of convenience, the four motions are called *basis motions*, and the torque-generation method is called *basis-motion torque composition (BMC)*. The details of algorism of the BMC are discussed in the next section.

III. BASIS-MOTION TORQUE COMPOSITION (MATHEMATICAL ANALYSIS)

In order to verify the proposition described in the previous section, the algorism of BMC is presented. By referring to eq.(5), the feedforward torque $\boldsymbol{\tau}_x = (\tau_{x1}, \tau_{x2})^{\mathrm{T}}$ for realizing the desired motion $\boldsymbol{q}_x(r_x(t))$ in eq.(14) follows

$$\begin{cases} \tau_{x1} = m_x a_{11}\eta_x'' + (2m_x + n_x)c_{x2}a\eta_x'' \\ \qquad +n_x a_{22}\eta_x'' - (2m_x n_x + n_x^2)s_{x2}a\eta_x'^2 \\ \qquad +m_x d_1 \eta_x' + \rho_1 \mathrm{sgn}(m_x \eta_x') \\ \tau_{x2} = m_x c_{x2} a\eta_x'' + (m_x + n_x)a_{22}\eta_x'' \\ \qquad +m_x^2 s_{x2} a\eta_x'^2 + n_x d_2 \eta_x' + \rho_2 \mathrm{sgn}(n_x \eta_x') \end{cases} \tag{17}$$

where $s_{x2} = \sin q_{x2}$, and $c_{x2} = \cos q_{x2}$. Then, by substituting eq.(15) into eq.(17), it can be transformed into

$$\begin{cases} \tau_{x1} = m_x \alpha_x^2 a_{11}\ddot{\eta} + (2m_x + n_x)\alpha_x^2 c_{x2} a\ddot{\eta} \\ \qquad +n_x \alpha_x^2 a_{22}\ddot{\eta} - (2m_x n_x + n_x^2)\alpha_x^2 s_{x2} a\dot{\eta}^2 \\ \qquad +m_x \alpha_x d_1 \dot{\eta} + \mathrm{sgn}(m_x \alpha_x)\rho_1 \mathrm{sgn}(\dot{\eta}) \\ \qquad -m_x \alpha_x^3 \beta_x a_{11}\dot{\eta} - (2m_x + n_x)\alpha_x^3 \beta_x c_{x2} a\dot{\eta} \\ \qquad -n_x \alpha_x^3 \beta_x a_{22}\dot{\eta} \\ \tau_{x2} = m_x \alpha_x^2 c_{x2} a\ddot{\eta} + (m_x + n_x)\alpha_x^2 a_{22}\ddot{\eta} \\ \qquad +m_x^2 \alpha_x^2 s_{x2} a\dot{\eta}^2 + n_x \alpha_x d_2 \dot{\eta} + \mathrm{sgn}(n_x \alpha_x)\rho_2 \mathrm{sgn}(\dot{\eta}) \\ \qquad -m_x \alpha_x^3 \beta_x c_{x2} a\dot{\eta} - (m_x + n_x)\alpha_x^3 \beta_x a_{22}\dot{\eta} \end{cases} \tag{18}$$

The expression (18) means that the desired feedforward torque can be described on the basis of the original motion primitive $\eta(t)$ in eq.(6) instead of the time-scale transformed motion primitive $\eta_x(r_x(t))$. In eq.(18), the parameters m_x, n_x, s_{x2}, c_{x2}, α_x, β_x, and η are preliminarily specified and known, but the other parameters (dynamics parameters) are unknown. Now, eq.(18) can be rewritten as

$$\boldsymbol{\tau}_x = N_{x1}(t)\boldsymbol{p}(t) + N_{x2}(t)\int_0^t \boldsymbol{p}(\xi)\,\mathrm{d}\xi \tag{19}$$

where

$$N_{x1}(t) = \begin{bmatrix} m_x\alpha_x^2 & (2m_x+n_x)\alpha_x^2 c_{x2} & n_x\alpha_x^2 & -(2m_x n_x + n_x^2)\alpha_x^2 s_{x2} & m_x\alpha_x & 0 & \mathrm{sgn}(m_x\alpha_x) & 0 \\ 0 & m_x\alpha_x^2 c_{x2} & (m_x+n_x)\alpha_x^2 & m_x^2\alpha_x^2 s_{x2} & 0 & n_x\alpha_x & 0 & \mathrm{sgn}(n_x\alpha_x) \end{bmatrix} \tag{20}$$

$$N_{x2}(t) = \begin{bmatrix} -m_x\alpha_x^3\beta_x & -(2m_x+n_x)\alpha_x^3\beta_x c_{x2} & -n_x\alpha_x^3\beta_x & 0 & 0 & 0 & 0 & 0 \\ 0 & -m_x\alpha_x^3\beta_x c_{x2} & -(m_x+n_x)\alpha_x^3\beta_x & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{21}$$

$$\boldsymbol{p}(t) = \begin{bmatrix} a_{11}\ddot{\eta} & a\ddot{\eta} & a_{22}\ddot{\eta} & a\dot{\eta}^2 & d_1\dot{\eta} & d_2\dot{\eta} & \rho_1\mathrm{sgn}(\dot{\eta}) & \rho_2\mathrm{sgn}(\dot{\eta}) \end{bmatrix}^{\mathrm{T}} \tag{22}$$

Note that the matrices $N_{x1}(t)$ and $N_{x2}(t)$ are known because they are composed of the only command parameters but the vector $\boldsymbol{p}(t)$ is unknown because it contains the uncertain dynamics parameters. If the unknown vector $\boldsymbol{p}(t)$ is derived from the dataset of basis-motion torque, the desired feedforward torque $\boldsymbol{\tau}_x$ can be formed.

Now, for adequately given m_i, n_i, q_{i01}, q_{i02}, and $r_i(t)$ $(i = a, \ldots, d)$, four motions

$$\begin{cases} (a): q_{a1}(r_a(t)) = m_a\eta_a(r_a(t)) + q_{a01}, & q_{a2}(r_a(t)) = n_a\eta_b(r_a(t)) + q_{a02} \\ (b): q_{b1}(r_b(t)) = m_b\eta_b(r_b(t)) + q_{b01}, & q_{b2}(r_b(t)) = n_b\eta_b(r_b(t)) + q_{b02} \\ (c): q_{c1}(r_c(t)) = m_c\eta_c(r_c(t)) + q_{c01}, & q_{c2}(r_c(t)) = n_c\eta_c(r_c(t)) + q_{c02} \\ (d): q_{d1}(r_d(t)) = m_d\eta_d(r_d(t)) + q_{d01}, & q_{d2}(r_d(t)) = n_d\eta_d(r_d(t)) + q_{d02} \end{cases} \tag{23}$$

are chosen as basis motions. The linear functions are chosen for the time-scale functions ($r_i(t) = \gamma_i t$ $(i = a, \ldots, d)$ for positive constants γ_i), and the time-scale transformed motion primitives $\eta_i(r_i(t))$ $(i = a, \ldots, d)$ are defined in the same manner as $\eta_x(r_x(t))$. Then, all of the feedforward torque for realizing the basis motions are obtained by the ILC. The obtained basis-motion feedforward torque can be described in the form of eq.(18) by replacing the subscript x in eq.(18) with $a, \ldots d$, respectively. Note that the terms including β_i $(i = a, \ldots, d)$ disappear in these expressions because the linear time-scale functions are chosen (*i.e.*, $\beta_i = 0$ $(i = a, \ldots, d)$). Hence, each feedforward torque is regarded as the linear equation of elements of $\boldsymbol{p}(t)$. Thus, over the time duration $t \in [0, T]$, the basis-motion torque can be described in the form

$$\boldsymbol{\tau}_{BM} = B(t)\boldsymbol{p}(t) \tag{24}$$

where

$$\boldsymbol{\tau}_{BM} = \begin{bmatrix} \boldsymbol{\tau}_a \\ \boldsymbol{\tau}_b \\ \boldsymbol{\tau}_c \\ \boldsymbol{\tau}_d \end{bmatrix}, \quad B(t) = \begin{bmatrix} N_{a1}(t) \\ N_{b1}(t) \\ N_{c1}(t) \\ N_{d1}(t) \end{bmatrix}, \tag{25}$$

$\boldsymbol{\tau}_a, \ldots, \boldsymbol{\tau}_d \in \Re^2$ denote the feedforward torque for realizing the basis motions, and $N_{a1}(t), \ldots, N_{d1}(t) \in \Re^{2\times 8}$ are defined by replacing the subscript x in eq.(20) with $a, \ldots d$, respectively. If the basis motions are adequately chosen so that the matrix $B(t)$ does not degenerate over $[0, T]$, then the inverse of $B(t)$ can be derived. Hence, multiplying eq.(24) by the inverse of $B(t)$ from the left-hand, we obtain

$$\boldsymbol{p}(t) = B^{-1}(t)\boldsymbol{\tau}_{BM} \tag{26}$$

Thus, the unknown vector $\boldsymbol{p}(t)$ can be derived from the basis-motion torque. Note that the matrix $B(t)$ is known because it is composed of the only parameters for the basis motions. Then, substituting eq.(26) into eq.(19) yields

$$\begin{aligned} \boldsymbol{\tau}_x = {} & N_{x1}(t)B^{-1}(t)\boldsymbol{\tau}_{BM} \\ & + N_{x2}(t)\int_0^t B^{-1}(\xi)\boldsymbol{\tau}_{BM}\,\mathrm{d}\xi \end{aligned} \tag{27}$$

Note that N_{x1} and N_{x2} are the known matrices for the desired motion and $B(t)$ is the known matrix for the basis motions.

Consequently, it is concluded that if the basis motions are chosen adequately so that $B(t)$ does not degenerate over

TABLE I
PARAMETER SETTINGS FOR THE BASIS MOTIONS

(i)	$r_i(t)$	m_i	n_i	q_{i01} [deg]	q_{i02} [deg]
Motion (a)	t	1.0	1.4	-50.0	5.0
Motion (b)	t	1.0	1.0	-50.0	5.0
Motion (c)	$1.2t$	1.0	1.0	-50.0	5.0
Motion (d)	$1.5t$	1.0	1.0	-50.0	5.0

TABLE II
PARAMETER SETTINGS FOR THE DESIRED MOTIONS

	$r_x(t)$	m	n	q_{x01}	q_{x02}
Case 1	$0.275t^3 - 0.475t^2 + 1.1t$	0.8	1.2	-30 [deg]	10 [deg]
Case 2	$1.25t$	0.8	1.2	-30 [deg]	10 [deg]
Case 3	$0.275t^3 - 1.175t^2 + 2.5t$	0.8	1.2	-30 [deg]	10 [deg]
Case 4	$0.025t^3 - 0.075t^2 + 0.9t$	-0.9	-1.1	30 [deg]	-10 [deg]
Case 5	$0.025t^3 - 0.075t^2 + 0.9t$	-0.8	-1.2	30 [deg]	-10 [deg]

the time duration $t \in [0, T]$ then the feedforward torque for realizing the desired motion can be derived from the basis-motion dataset pairs (torque and joint trajectories) and the desired-motion joint trajectory as shown in eq.(27).

IV. EXPERIMENTS

In order to confirm the effectiveness of BMC, we conducted experiments using an industrial robot arm: PA-10 (Mitsubishi heavy industries, LTD.). The robot arm was placed horizontally and the base frame of robot was fixed at a base as shown in Fig.4. The robot arm has seven joints in total, but the only two joints are used and the other joints are locked by mechanical breaks. The controller of PA-10 allows two control modes: “velocity mode” and “torque mode.” Since our concern is to generate torque for control, we selected the torque mode.

The motion primitive of eq.(6) was set with $\phi = 80$[deg] and $T = 2.0$[s], and the four basis motions of eq.(23) were determined by the parameters in TABLE I. The online feedback gains in eq.(2) were set with $K_v = \mathrm{diag}(1.4, 0.4)[\mathrm{Nms/rad}]$ and $K_p = \mathrm{diag}(23.0, 2.9)[\mathrm{Nm/rad}]$. The feedback gains, as discussed above, were set so that the feedback input can be sufficiently smaller than the feedforward input. Under the condition, all of the feedforward torque for realizing the basis motions were obtained by the ILC.

Using the BMC, we attempted to generate five test motions shown in TABLE II. The desired feedforward torque was derived from eq.(27) off-line, and the derived torque was applied to the controller in eq.(2), respectively. The feedback gains in this step were set with the same values as those in the step for the generation of basis-motion torque so as not to change the properties of dynamics.

Figure 5 depicts the transient responses of joint angles and angular velocities in Cases 1, 2, and 3 where the motion-scale parameters (m_x, n_x) were fixed but the different time-scale functions ($r_x(t)$) were chosen (the terminal time is same in the three cases: $T_x = 2.5$[s]). As shown in Fig.5, the trajectories of both angles and angular velocities are nearly coincident with the corresponding desired trajectories even in the case of the different time-scale functions, though slight oscillations are observed in the low-speed areas of angular-velocity profiles. Figure 6 depicts the torque profiles of feedforward and feedback inputs in Case 1. We, in section

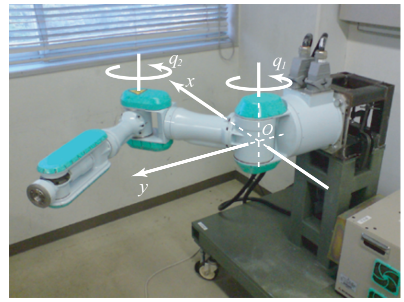
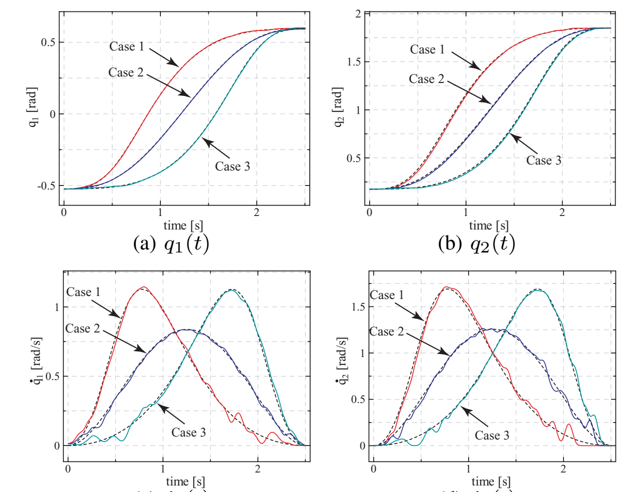


Fig. 4. An experimental setup

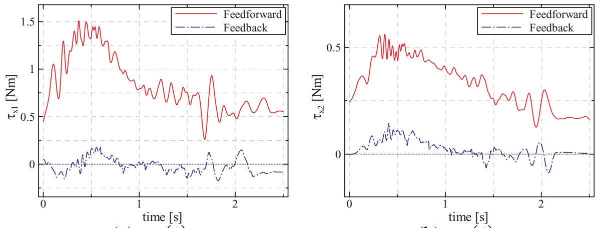


(a) $q_1(t)$ (b) $q_2(t)$
(c) $\dot{q}_1(t)$ (d) $\dot{q}_2(t)$

Fig. 5. Experimental results in Cases 1, 2, and 3 (in the case of the different time-scale functions)

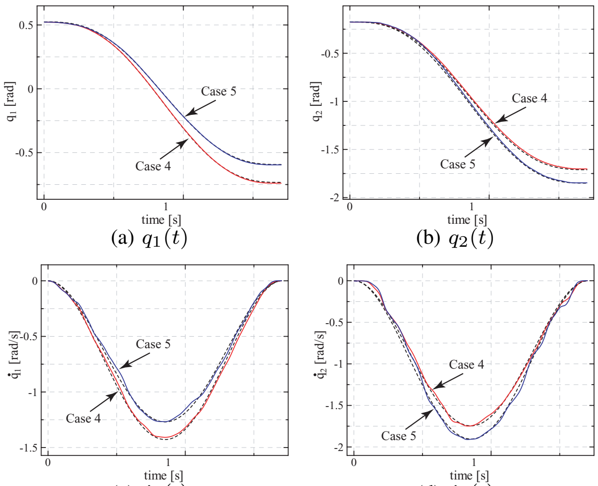
2, supposed that the feedback gains are chosen so that the feedback effect can be significantly smaller than the feedforward effect. Figure 6 demonstrates that the feedforward inputs are dominant in the controller.

Figure 7 depicts the joint-angle profiles and the joint-angular-velocity profiles in Cases 4 and 5 where the time-scale function is fixed but the different motion-scale parameters are chosen (the terminal time is same in the two cases but it is shorter than that in Fig.5: $T_x = 1.7$[s]). In the cases, the negative motion-scale parameters were set, that is, the direction of motions with the negative motion-scale parameters is different from that of basis motions. Regardless of the condition, the tracking performance in the two cases is better than that in the three cases of Fig.5. Furthermore, Fig.8 depicts the error norm of angular velocities by the BMC in Case 4 in comparison to that by the computed torque method (CTM). In order to apply the CTM to the system, the dynamics in eq.(5) were supposed and all of the dynamics parameters were evaluated by the online dynamic-parameter estimation. Then, the feedforward torque generated by the CTM was applied to the controller in eq.(2). Figure 8 demonstrates that the error norm of BMC is as a whole smaller than that of CTM. Regardless of the assumption of the same dynamics structure, the velocity error norm of BMC was kept small in comparison to that of CTM. Differently from the CTM based on determining every dynamics parameter, the BMC is based on four arithmetical operations of time-series data of basis-motion torque. It is considered that



(a) $q_1(t)$ (b) $q_2(t)$

Fig. 6. Torque profiles of feedforward and feedback inputs in Case 1



(a) $q_1(t)$ (b) $q_2(t)$

(c) $\dot{q}_1(t)$ (d) $\dot{q}_2(t)$

Fig. 7. Experimental results in Cases 4 and 5 (in the case of the different motion-scale parameters (m_x and n_x))

such an operation contributes the enhancement of robustness to modeling errors of dynamics. Thus, these experimental results support the effectiveness of BMC.

In further investigations, we found that the tracking performance of motions generated by the BMC tended to be bad when the magnitude of $\beta_x(t)$ became large during movements. The problem was caused by the noise amplification incurred by arithmetical operations of time-series torque data. We confirmed in simulations that this was not an algorithmic issue. Hence, the filtering of basis-motion torque data is important in the BMC. If an adequate filtering is developed, the problem will be solved.

As shown in Fig.5, the oscillations were observed in the low-speed areas of less than 0.25[rad/s]. Such oscillations were observed even in the case of the CTM. It is considered that the oscillations appeared since the assumed friction model did not fit to the actual one. As discussed in the book written by Armstrong-Hélouvry [3] and the paper written by Kennedy and Desai [4], joint friction of a robot arm is often formulated by the Stribeck model. On the other hand, our joint-friction model is assumed as a linear model in eq.(5) to avoid theoretical discussion be complicated. The notable characteristic of Stribeck model is a good fitting in low speed. Hence, if the joint friction is formulated by the Stribeck model, the oscillations in the velocity profiles will be improved. The reasonability of consideration is supported by the experimental result that the tracking performance of fast motion in Fig.7 is better than that of slow motion in Fig.5.

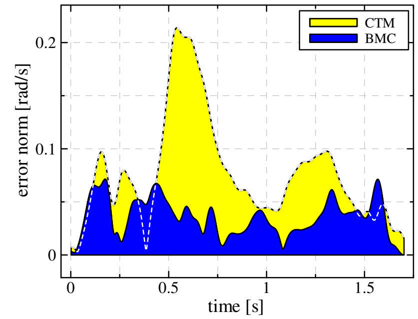


Fig. 8. Error norm of angular velocities in Case 4 (the basis-motion torque composition versus the computed torque method)

V. CONCLUSIONS

The basis-motion torque composition, which allows to generate a motion to a specified final posture according to a specified time profile based on arithmetic operations of time-series torque data of motions related to a unique motion primitive, was suggested. Then, the torque-generation algorism, the sufficient condition for the choice of basis motions, and the experimental results were presented. The class of generable motions is limited even in the case of the BMC. Extension to the case of a different time-scale function at each joint will be discussed in the future work.

REFERENCES

[1] C. G. Atkeson, C. H. An, and J. M. Hollerbach, "Estimation of inertial parameters of manipulator loads and links," *International Journal of Robotics Research*, vol. 5, no. 3, pp. 101–118, 1986.

[2] P. K. Khosla, "Estimation of robot dynamics parameters: theory and application," *Int. J. of Rob. and Autom.*, vol. 3, no. 1, pp. 35–41, 1988.

[3] B. Armstrong-Hélouvry, *Control of Machines with Friction*. MA: Kluwer Academic Publishers, 1991.

[4] C. W. Kennedy and J. P. Desai, "Modeling and control of the mitsubishi PA-10 robot arm harmonic drive system," *IEEE/ASME Transactions on Mechatronics*, vol. 10, no. 3, pp. 263–274, 2005.

[5] M. Garden, "Learning control of actuators systems," *U.S. Patent*, no. 3555252, 1971.

[6] M. Uchiyama, "Formation of high-speed motion pattern of a mechanical arm by trial," *Transactions of the Society of Instrument and Cotrol Engineerings*, vol. 14, no. 6, pp. 706–712, 1978, (in Japanese).

[7] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *J. of Rob. Sys.*, vol. 1, no. 2, pp. 123–140, 1984.

[8] S. Arimoto, *Control Theory of Non-linear Mechanical Systems: A Passivity-based and Circuit-theoretic Approach*. Oxford, UK: Oxford Univ. Press, 1996.

[9] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Systems Magazine*, vol. 26, no. 3, pp. 96–114, 2006.

[10] S. Arimoto, M. Sekimoto, and S. Kawamura, "Task-space iterative learning for redundant robotic systems: Existence of a task-space control and convergence of learning," *SICE Journal of Control, Measurement, and System Integration*, vol. 1, no. 4, pp. 312–319, 2008.

[11] M. Sekimoto, S. Arimoto, and S. Kawamura, "Task-space iterative learning for redundant robots: Simultaneous acquirements of desired motion and force trajectories under constraints," in *Proc. of the 9th Int. IFAC Sym. on Robot Control*, Gifu, Japan, Sept. 2009, (in press).

[12] S. Kawamura, F. Miyazaki, and S. Arimoto, "Intelligent control of robot motion based on learning method," in *Proc. of the IEEE Int. Symp. on Intell. Control*, Washington, DC, Jan. 1987, pp. 365–370.

[13] S. Kawamura and N. Sakagami, "Planning and control of robot motion based on time-scale transformation," in *Advances in Robot Control*, S. Kawamura and M. Svinin, Eds. Springer-Verlag, 2006, pp. 157–178.

[14] M. Sekimoto, S. Kawamura, T. Ishitsubo, S. Akizuki, and M. Mizuno, "Posture control of a multi-joint robot based on composition of feedforward joint-torques acquired by iterative learning," in *Proc. of the ICROS-SICE Int. Joint Conf. 2009*, Japan, Aug. 2009, (in press).