2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 1-5, 2023. Detroit, USA

An Approach to Design a Biomechanically-Inspired Reward Function to Solve a Patience Cube under Reinforcement Learning Framework*

Janghyeon Kim[1], Ho-Jin Jung[1], Dae Han Sim[1], Ji-Hyeon Yoo[1], Song Woo Kim[1], and Han Ul Yoon[2]

Abstract— **This paper presents an approach to design a reward function by adopting both control theoretic and biomechanical perspectives. In reinforcement learning (RL), a reward function plays a crucial role for an RL agent training; especially, a task learning time and a task performance. Accordingly, designing a reward function becomes a key issue to train an RL agent generating human-like policy/strategy to perform dexterous manipulation. Since human beings are good at producing heuristic approaches to complete a given task, determining a set of basis functions as well as corresponding weights used not to be so straightforward. In this study, we consider solving a patience cube as an example of a dexterous manipulation task. In our approach, we first employed a quadratic regulator form as a backbone of a desired reward function. Next, the kinematic data of a controlled object and the sEMG data of a human expert were measured while performing a demonstration to solve a patience cube. Then, from the measured data, the weights of the basis functions were determined by utilizing muscle synergy extraction and inverse optimal control as two key tools. Finally, an RL agent was trained by the designed reward function and comparative analysis versus the other RL agents trained by prototypical weight settings was followed. The result showed that the RL agent trained by our approach yielded human-like learning curve as well as policy successfully and outperformed the others in terms of a task success rate and a task completion time. These findings substantiated the feasibility of extending our approach to an assistive robotic manipulator or prosthesis design to perform the activities of daily living.**

I. INTRODUCTION

Human beings perform various dexterous manipulation tasks in daily living activities such as grasping a coffee mug, opening a bottle cap, twisting a door knob, handling a tool, and so on. Reinforcement learning (RL) has shown successful outcomes for the aforementioned dexterous manipulation tasks, e.g., object twirling [1], in-hand object reorientation [2], rotating a cross-shaped valve [3], different grasp types [4], picking up a hammer to drive a nail [5], solving Rubik's cube [6], etc.

Existing RL studies have been disseminated reward functions according to various given tasks. Levine et al. showed that a quadratic regulator (QR) form could serve as a backbone of a reward function to train an RL agent for robotic dexterous manipulation and presented successful results for stacking blocks and screwing a cap on a water bottle [7]. They also proposed an end-to-end approach that maps raw image observations directly to torques at the robot's motors for visuo-motor tasks [8]. Duan et al. presented a task-specific reward design to solve continuous control problems such as mountain car, simple humanoid, and maze, which can be served as a benchmark [9].

By following a footprint in RL, Todorov and Jordan found and prelusorily initiated that human behavior while performing tasks related to visuo-motor coordination could be well-explained by the reward function of a QR form [10]. Certainly, dexterous manipulation is a task which requires visuo-motor coordination; therefore, we can expect that the QR form can also serve as a backbone to design a reward function in perspective of the RL unless the task is highly specific. Nevertheless, determining the corresponding weights for the quadratic basis functions still remains as a problematic issue. Especially, for a given task, human beings produce policy/strategy by heuristic approach, which is optimal in some sense (i.e., energy consumption). This characteristic nature of human behavior even makes the problem of designing a reward function more difficult in sense of optimality.

Inverse optimal control (IOC) has been utilized to solve the problem of a reward/cost function design which enables the trained agent to imitate human behavior [11], [12]. By regarding the human demonstration as an approximately optimal control input to (as well as resulting state of) a controlled system, the IOC can infer the reward/cost function by finding the corresponding weight vectors of the basis functions when the forms of basis functions are known [13], [14]. Considering dexterous manipulation in perspective of a task related to visuo-motor coordination, we can think of a motor activity as a primitive version of a state feedback control based on visual information containing a controlled object state [8], [15]. Therefore, if we find a mapping or a relationship between human expert's motor activity and a control input and employ it to design a reward function, then the trained RL agent can be endowed with somewhat human-like optimality criteria to perform dexterous manipulation tasks.

In this paper, we propose an approach to design a reward function which enables an RL agent to perform the dexterous manipulation tasks with human-like optimality criteria. Specifically, we define the reward function as a function of the state of a controlled object and the motor activity of a human expert. We note that solving a patience cube will be considered as an

*This work was partially supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT) (Grant No. 2021R1F1A1063339) and by Industrial Technology Innovation Program funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea) (Grant No. 20023014).

[1]Janghyeon Kim, Ho-Jin Jung, Dae Han Sim, Ji-Hyeon Yoo, Song Woo Kim are with the Department of Computer Science, Yonsei University (Mirae), Wonju, Gangwon 26493, Korea {janghyeonk,hojinj dhsim,jihyeonyoo,swkim}@yonsei.ac.kr

[2]Han Ul Yoon is with Faculty of the Division of Software, Yonsei University (Mirae), Wonju, Gangwon 26493, Korea huyoon@yonsei.ac.kr

2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-6654-9190-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/IROS55552.2023.10341831

example of dexterous manipulation task throughout this paper. The surface electromyography (sEMG) of a pilot subject and the kinodynamic data of the patience cube were measured via an armband type sensor and an inertial measurement unit (IMU) while solving the patience cube. From sEMG data being measured, we first extracted modularized muscle groups, which are usually referred to as muscle synergies, and corresponding activation curves. Next, we mapped the motor activities of the pilot subject onto the mechanical torques of 2-DoF robotic wrist-hand by identifying a relationship between the activation curve of the muscle synergy and the angular velocity/acceleration of the cube. Then, a reward function was designed as a function of the patience cube state and human expert's motor activity. Finally, the RL agent was trained by the designed reward function and its performance was evaluated by comparative analysis. To our best knowledge, the idea akin to the proposed approach has not yet been fully accounted for. In existing research, sEMG has been often employed as an input signal to control robotic manipulator [16] or robotic prosthesis [17], [18]. In our approach, instead, sEMG data as well as kinodynamic data were utilized to design a reward function yielding a human-like optimal policy to control such robotic systems.

The rest of the paper is organized as follows: our approach to design a reward function is introduced in Sec. II. In Sec. III, the experiment to train and evaluate the robotic agent for both the simulation world problem and the real world problem is presented. The results are reported, investigated by comparative analysis, and discussed in Sec. IV. Sec. V will be the conclusion and future work of this paper.

II. APPROACH TO DESIGN A REWARD FUNCTION IN BIOMECHANICAL PERSPECTIVE

A. Patience Cube

Fig. 1 shows a patience cube which will be considered as an example of dexterous manipulation task throughout this paper. Fig. 1(a) presents the appearance of the patience cube, and Fig. 1(b) and Fig. 1(c) show the patience cube in a human hand in real world and a 2-DoF robotic wrist-hand in simulation world, respectively.

For this patience cube, it is regarded as "solved" when an iron ball is located into a center hole so that the iron ball is stuck and does not move. There exist 24 different types of patience cubes for which a highly dexterous hand manipulation as well as patience is required to solve them.

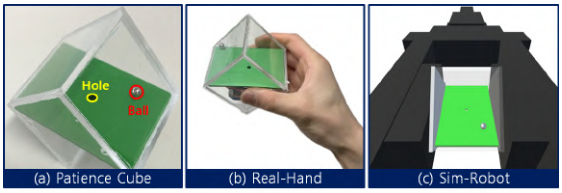

Fig. 1. Patience cube and real/virtual world examples: (a) the appearance, (b) the patience cube in a human hand in real world, and (c) the patience cube in a 2-DoF robotic wrist-hand in simulation world.

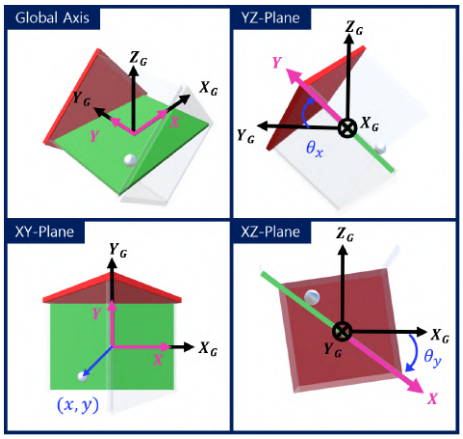

Fig. 2. A coordinate frame to represent the movement of ball and plate and derive a dynamic model.

Since the main purpose of this study is to design a reward function which enables an RL agent to generate human-like policy/strategy under RL framework; hence, we start by challenging rather easier one.

B. The Dynamic Model of a Patience Cube: Ball and Flat-Plate Model

Fig. 2 depicts a coordinate frame to represent the movement of ball and plate and derive a dynamic model. Parameters are defined in Table I. Let x and θ_y be a distance from a center hole with respect to x-axis and a rotation angle along y-axis, respectively (y and θ_x are defined by the same convention). By following the derivation in [19], we obtain

$$\begin{aligned}
&\ddot{x} + \frac{m_b g}{J_b/r_b^2 + m_b}\sin\theta_y - \frac{m_b x}{J_b/r_b^2 + m_b}\dot{\theta}_y^2 - \frac{F_{\mu,x}}{J_b/r_b^2 + m_b} = 0 \\
&\ddot{\theta}_y + \frac{2m_b x\dot{x}}{m_b x^2 + J_{p,y}}\dot{\theta}_y + \left(\frac{m_b g x + m_p g\frac{l_x}{2}}{m_b x^2 + J_{p,y}}\right)\cos\theta_y = \tau_y \\
&\ddot{y} + \frac{m_b g}{J_b/r_b^2 + m_b}\sin\theta_x - \frac{m_b y}{J_b/r_b^2 + m_b}\dot{\theta}_x^2 - \frac{F_{\mu,y}}{J_b/r_b^2 + m_b} = 0 \\
&\ddot{\theta}_x + \frac{2m_b y\dot{y}}{m_b y^2 + J_{p,x}}\dot{\theta}_x + \left(\frac{m_b g y + m_p g\frac{l_y}{2}}{m_b y^2 + J_{p,x}}\right)\cos\theta_x = \tau_x
\end{aligned} \tag{1}$$

TABLE I
THE DEFINITION OF PARAMETERS

Name	Symbol	Value	Unit
Ball mass	m_b	1.057×10^{-3}	kg
Ball radius	r_b	3.175×10^{-3}	m
The moment of inertia of the ball	J_b	4.252×10^{-9}	kg· m^2
Plate mass	m_p	1.00×10^{-4}	kg
Plate length	l_x and l_y	51.14×10^{-3} and 35.73×10^{-3}	m
The moment of inertia of the plate	$J_{p,x}$ and $J_{p,y}$	1.064×10^{-8} and 2.179×10^{-8}	kg· m^2
Coefficient of friction	μ	0.3604	N/A

where τ_y and τ_x represent applied torques along y- and x-axis, respectively. In (1), $F_{\mu,x}$ and $F_{\mu,y}$ are frictional forces between the ball and the plate along x- and y-axis directions:

$$\begin{aligned} F_{\mu,x} &= -\mu m_b g \cos\theta_y \text{sign}(\dot{x}) \\ F_{\mu,y} &= -\mu m_b g \cos\theta_x \text{sign}(\dot{y}). \end{aligned} \tag{2}$$

From (1), we define a state s as

$$s = [x, \dot{x}, \theta_y, \dot{\theta}_y, y, \dot{y}, \theta_x, \dot{\theta}_x]^T.$$

Since τ_y and τ_x generate the movement of the ball along x- and y-direction, respectively; we define a control input a as

$$a = [a_x, a_y]^T = [\tau_y, \tau_x]^T$$

to represent the direction of the ball movement and the corresponding actions explicitly. Now, (1) can be represented as $\dot{s} = f(s, a)$; accordingly, let t_s be a sampling time, we have a discrete time version

$$s_{k+1} = s_k + t_s f(s_k, a_k) \tag{3}$$

in which

$$\begin{aligned} s_{1,k+1} &= s_{1,k} + t_s s_{2,k} \\ s_{2,k+1} &= s_{2,k} + t_s \frac{(-m_b g \sin s_{3,k} + m_b s_{1,k} s_{4,k}^2 + F_{\mu,x})}{J_b/r_b^2 + m_b} \\ s_{3,k+1} &= s_{3,k} + t_s s_{4,k} \\ s_{4,k+1} &= s_{4,k} + t_s [a_{x,k} \\ &\quad + \frac{(-2m_b s_{1,k} s_{2,k} s_{4,k} - (m_b g s_{1,k} + m_p g \frac{l}{2}) \cos s_{3,k})}{m_b s_{1,k}^2 + J_{p,y}}] \\ s_{5,k+1} &= s_{5,k} + t_s s_{6,k} \\ s_{6,k+1} &= s_{6,k} + t_s \frac{(-m_b g \sin s_{7,k} + m_b s_{5,k} s_{8,k}^2 + F_{\mu,y})}{J_b/r_b^2 + m_b} \\ s_{7,k+1} &= s_{7,k} + t_s s_{8,k} \\ s_{8,k+1} &= s_{8,k} + t_s [a_{y,k} \\ &\quad + \frac{(-2m_b s_{5,k} s_{6,k} s_{8,k} - (m_b g s_{5,k} + m_p g \frac{l}{2}) \cos s_{7,k})}{m_b s_{5,k}^2 + J_{p,x}}]. \end{aligned} \tag{4}$$

C. Problem Definition: Design a Reward Function from a Human Expert Demonstrated Biomechanical Data

Let the sequence of the state and control input of/to the patience cube be

$$s = \{s_0, s_1, \cdots, s_{N-1}\} \quad \text{and} \quad a = \{a_0, a_1, \cdots, a_{N-1}\}$$

for $k = 0, \cdots, N$, respectively. Now, throughout this study, we consider the following reward functional, $G(s, a)$, of a specific form:

$$G(s,a) = \sum_{k=0}^{N-1} r(s_k, a_k) = \sum_{k=0}^{N-1} -\left[a_k^T R a_k + s_k^T Q s_k\right] \tag{5}$$

where

$$R = \begin{bmatrix} c_{ax} & 0 \\ 0 & c_{ay} \end{bmatrix}, Q = \begin{bmatrix} c_p & 0 & 0 & 0 & \cdots & 0 \\ 0 & c_v & 0 & 0 & \cdots & 0 \\ 0 & 0 & c_\theta & 0 & \cdots & 0 \\ 0 & 0 & 0 & c_\omega & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & c_\omega \end{bmatrix}. \tag{6}$$

Recall that the ball and flat-plate model discussed in Sec. II-B. The subscripted parameter c_{ax} and c_{ay} represent the weights for control effort to rotate the patience cube along x- and y-axis, respectively. c_p, c_v, c_θ, and c_ω represent the weights for a positional deviation between the ball and the center hole of the patience cube, the velocity of the ball, an angular deviation of the plate, and the angular velocity of the plate.

For disambiguation, from now on, we refer to the following problem as a *forward optimal control problem*:

$$\begin{aligned} \max_a \quad & G(s,a) = \sum_{k=0}^{N-1} -\left[a_k^T R a_k + s_k^T Q s_k\right] \\ \text{sub. to } & s_{k+1} = s_k + t_s f(s_k, a_k) \\ & s_0 = s_{\text{init}} \\ & s_N = s_{\text{goal}}. \end{aligned} \tag{7}$$

Given the set of tuples (s^*, a^*) (which is assumed to be approximately optimal solution), in contrast, an *inverse optimal control (IOC) problem* is to infer a reward functional with unknown parameters, e.g., c_{ax} through c_ω in (5) and (6); namly, we want to solve

$$\text{Given } (s^*, a^*) \longrightarrow \text{Infer } G(s,a). \tag{8}$$

Now, let a_h be a human effort (in biomechanical perspective) to manipulate the patience cube and a be a control input (torque) as it was. In this study, we also want to identify a relationship (or a mapping), denoted by $\varphi(\cdot)$, which performs

$$\varphi(a_h) \to a. \tag{9}$$

In sum, we can design the biomechanically-inspired reward function, which is expected to enable the trained RL agent to generate human-like policy/strategy, by the following procedure

$$\text{Given } (s^*, a_h^*) \to (s^*, \varphi(a_h^*)) \to (s^*, a^*) \to \text{Infer } G(s,a),$$

which will eventually give us an explicit form of $r(s_k, a_k)$. Hence, our problem definition can be summarized as follows:

Given (s^*, a_h^*),
Identify $\varphi(a_h^*) \to a$ by investigating sEMG, then
Infer $G(s,a)$ by estimating unknown coefficients.

D. Extraction of Muscle Synergy and Corresponding Activation Curve

Bernstein found that a central nervous system does not control individual muscles, but activates modularized muscle groups, which are usually called muscle modules or muscle

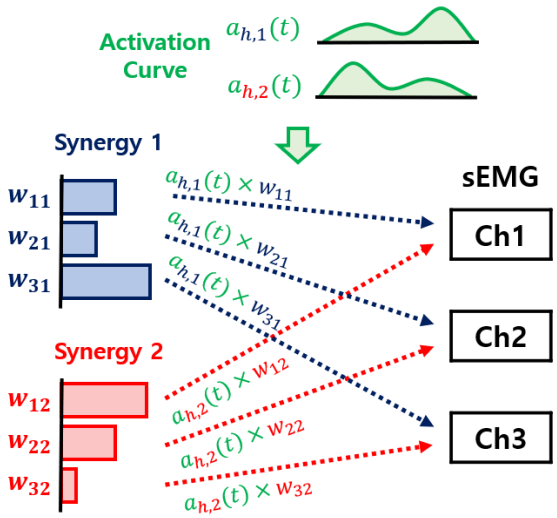

Fig. 3. An illustrative example of the muscle synergy w and the activation curve a_h (in case of the number of channel $l = 3$ and the number of synergies $n = 2$).

synergies, to perform a given specific task [20]. Many follow-up studies have been substantiated the existence of muscle synergies for maintaining a postural balance [21], [22], arm force and stiffness control [23], [24], swimming [25], [26], walk and balance [27], [28], etc. Moreover, the muscle synergies have been utilized to control a robotic manipulator as well as a robotic prosthesis [18], [29].

The muscle synergy can be defined mathematically as follows. Let $M \in \mathbb{R}^{l\times k}$ be sEMG data containing l-EMG channels during k-time steps. Also, let $w_i = [w_{1i}, \cdots, w_{li}]^T \in \mathbb{R}^{l\times 1}$ and $a_{h,i} = [a_{h,i1}, \cdots, a_{h,ik}] \in \mathbb{R}^{1\times k}$ represent i^{th} muscle synergy and the corresponding activation curve, respectively. Now, the sEMG data M can be expressed by

$$M = \sum_{i=1}^{n} w_i \times a_{h,i} \tag{10}$$

where n is the number of existing muscle synergies. w_i and $a_{h,i}$ are normally extracted by solving a non-negative matrix factorization (NNMF) [30], [21]. We note that the muscle synergy concept is depicted graphically in Fig. 3.

Berger and d'Avella have found that there exists a linear mapping from muscle synergy onto generated force during arm movement [23], [31]. Inouye and Valero-Cuevas have also found a linear relationship between the activation of muscle synergies and the stiffness of arm endpoint [24]. Hence, by following [23], [24], we also set a relationship between a_h and a as

$$\varphi(a_h) = \alpha a_h =: a \tag{11}$$

where α is a positive constant, which will be implicitly involved in the coefficients c_{ax} and c_{ay} of R matrix.

E. Applied Numerical Method to Solve the IOC Problem

To solve the IOC problem of (8), we begin with defining a discrete time Hamiltonian H_k

$$\begin{aligned} H_k(s_k, a_k, \lambda_k) &= -r(s_k, a_k) + \lambda_k^T f(s_k, a_k) \\ &= a_k^T R a_k + s_k^T Q s_k + \lambda_k^T f(s_k, a_k) \end{aligned} \tag{12}$$

where $\lambda_k \in \mathbb{R}^{8\times 1}$ is a costate vector at time step k. For clarity, we rewrite (12) as

$$H_k(s_k, a_k, \lambda_k) = c^T \phi(s_k, a_k) + \lambda_k^T f(s_k, a_k) \tag{13}$$

where

$$c = [c_{ax}, c_{ay}, c_p, c_v, c_\theta, c_\omega, c_p, c_v, c_\theta, c_\omega]^T \in \mathbb{R}_+^{(2+8)\times 1} \tag{14}$$

is a coefficient vector and $\phi(s_k, a_k) : \mathbb{R}^{(2+8)} \to \mathbb{R}_+^{(2+8)}$. Now, from the Minimum Principle, we know

$$\nabla_{s_k} H_k = \nabla_{s_k}\phi(s_k, a_k)\, c + \nabla_{s_k} f(s_k, a_k)\, \lambda_k = -\frac{\lambda_{k+1} - \lambda_k}{t_s}. \tag{15}$$

Then, by rearranging the two terms of (15) from the right side, we obtain the costate propagation equation

$$\lambda_{k+1} = -t_s \nabla_{s_k}\phi(s_k, a_k)\, c \; + \; [I - t_s \nabla_{s_k} f(s_k, a_k)]\; \lambda_k \tag{16}$$

where $I \in \mathbb{R}^{8\times 8}$ is an identity matrix. From a necessary condition for optimality, we also have

$$\nabla_{a_k} H_k = \nabla_{a_k}\phi(s_k, a_k)\, c \; + \; \nabla_{a_k} f(s_k, a_k)\; \lambda_k = \mathbf{0}, \tag{17}$$

thus,

$$\nabla_{a_k}\phi(s_k, a_k)\, c \; + \; \nabla_{a_k} f(s_k, a_k)\; \lambda_k = \mathbf{0}. \tag{18}$$

If we define a vector z_k as

$$z_k = [\; c^T,\; \lambda_{k+1}^T,\; \lambda_k^T \;]^T \in \mathbb{R}^{26\times 1}, \tag{19}$$

then, (16) and (18) can be combined into a system of equation

$$\begin{bmatrix} A_{11,k} & A_{12,k} & A_{13,k} & A_{14,k} \\ A_{21,k} & A_{22,k} & A_{23,k} & A_{24,k} \end{bmatrix} z_k =: A_k z_k \tag{20}$$

in which the submatrices are

$$\begin{aligned} &A_{11,k} = O \in \mathbb{R}^{8\times 2},\; A_{12,k} = -t_s \nabla_{s_k}\phi(s_k, a_k) \in \mathbb{R}^{8\times(4+4)}, \\ &A_{13,k} = I \in \mathbb{R}^{8\times 8},\; A_{14,k} = -I + t_s \nabla_{s_k} f(s_k, a_k) \in \mathbb{R}^{8\times 8}, \\ &A_{21,k} = \nabla_{a_k}\phi(s_k, a_k) \in \mathbb{R}^{2\times 2},\; A_{22,k} = O \in \mathbb{R}^{2\times(4+4)}, \\ &A_{23,k} = O \in \mathbb{R}^{2\times 8},\; A_{24,k} = \nabla_{a_k} f(s_k, a_k) \in \mathbb{R}^{2\times 8} \end{aligned} \tag{21}$$

where O represents zero matrix. Consequently, the dimension of A_k is $\mathbb{R}^{10\times 26}$.

The abovementioned method allows us to solve the given IOC problem of (8) by solving the following least square problem

$$\min_z \sum_{k=0}^{N-1} \|A_k^* z_k\|^2 \tag{22}$$

where A_k^* represents A_k evaluated with (s_k^*, a_k^*) at a time step k.

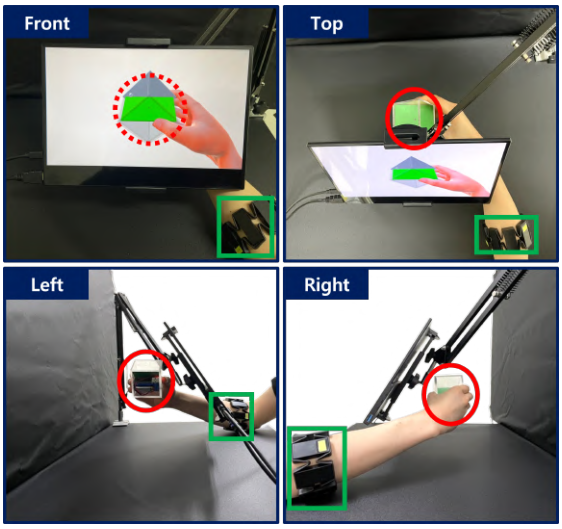

Fig. 4. The experimental setup to obtain human expert's data: a patience cube is an initial pose (dashed red circle), custom-made patience cube (solid red circle), and sEMG sensor (solid green square).

III. EXPERIMENTS

A. Human Subject and Experimental Setup

A healthy young adult(gender=male, age=25) was recruited for this study. The recruited subject gave an informed consent form prior to his participation and were instructed about the experimental protocol. The subject had a profession that involves manual work that requires dexterity. However, since the subject answered as "never experienced with a patience cube," he was instructed to have two weeks practice period with the patience cube being considered in this study. Consequently, the subject became able to solve the patience cube within the desired task-completion time (less than 15 second in average), which met "an expert" criterion for this study. This study was approved by the Institutional Review Board of the Yonsei University Mirae Campus (Approval No. 1041849-202201-BM-018-01).

Fig. 4 shows the experimental setup consists of a custom-made patience cube, a 13.3-inch display monitor mounted on a monitor arm, a office table, and an 8-channel armband type sEMG sensor (MyoArmband, Thalmic Labs, Brooklyn, NY, US). The subject was instructed to sit comfortable on an office chair at 30cm from the display monitor while maintaining his elbow on the table. The background wall area behind the display monitor as well as the table surface were covered with thin black paper to prevent distractions caused by the background.

B. Experimental Protocol

The experimental protocol can be summarized as follows:

- **Obtain** (s^*, a_h^*) from human expert's demonstration
- **Identify** $\varphi(a_h) =: a$ by extracting muscle synergies and corresponding activation curves
- **Design** a reward function $r(s, a)$ by solving the IOC problem
- **Train** the RL agent under the designed reward function

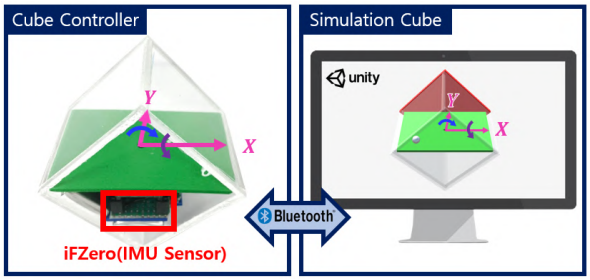

Fig. 5. The custom-made patience cube which was utilized as an input device to a rendered cube in virtual reality.

- **Evaluate** the task performance of the trained RL agent by comparative analysis

The above-mentioned protocol will be explained in detail below step-by-step.

1) Obtaining (s^*, a_h^*) from Human Expert's Demonstration: Fig. 5 presents the custom-made patience cube which was utilized as an input device to a rendered cube in virtual reality. To eliminate any potential reality gap between a simulation and a real environments (even when the derived ball and flat-plate model in Sec. II-B is perfect), we unified the experiment/train environment into the virtual environment for the subject/the RL agent. Namely, the subject tried to solve the patience cube being rendered in virtual environment.

As shown in Fig. 4, the subject was holding the custom-made patience cube behind the display monitor while watching the virtual cube being held by a phantom hand. The subject was instructed not to change a grip position during one trial to maximize synchronizing effect between two environments. The subject's right forearm was equipped with the sEMG sensor. The deployment of sEMG channels is presented in Table II.

TABLE II
THE DEPLOYMENT OF SEMG CHANNELS ON USER'S FOREARM

Channel No.	Ch #3 #4 #5	Ch #1 #7 #8	Ch #2 #6
Coverage Side	Radius Side	Ulna Side	Boundaries

A session consisted of 20 trials with pseudo-randomized initial ball position when each trial begins. A trial was programed to start automatically when the subject located the cube at the display center (see Fig. 4, dashed red circle) with $\theta_x \approx 0$ and $\theta_y \approx 0$. Sampling time was set to $t_s = 20$ms. The data is measured and stored rowwise in the following format: $(t_s, x, y, \dot{x}, \dot{y}, \ddot{x}, \ddot{y}, \theta_x, \theta_y, \dot{\theta}_x, \dot{\theta}_y, \ddot{\theta}_x, \ddot{\theta}_y, m_1, \cdots, m_8)$ where $m_1, \cdots, m_8$ represents the 8-channel sEMG data.

2) Identifying the Control Input by Extracting Muscle Synergies and Corresponding Activation Curves: After the subject completing the demonstration session (10 minute break time), the experiment was followed by a muscle synergy extraction session. A wrist joint was assumed to have 2-DoF. First, the subject was instructed to perform supination, pronation, upward, downward along x- and y-axis

TABLE III

DESIGNED REWARD FUNCTION AND CORRESPONDING WEIGHTS SETTING

Reward designed as/by	c_{ax}	c_{ay}	c_p	c_θ	c_v	c_ω
Cheap control	0.01	0.01	1	1	0.01	0.001
Balanced control	1	1	1	1	0.01	0.001
Expensive control	100	100	1	1	0.01	0.001
Ours*	**0.0201**	**0.0201**	**1**	**1.2538**	**0.0001**	**0.0004**

and positive/negative directions while holding the custom-made patience cube. Each action was repeated ten times. Since the four actions were repeated ten times, we obtained $4 \times 10 = 40$ sEMG data. Next, the measured sEMG was low-pass filtered with cutoff frequency 10Hz and intra-channel normalized. Finally, the muscle synergies and activation curves were extracted by NNMF introduced in Sec II-D. The number of muscle synergies was set to $n = 3$ based on the variance accounted for. We chose a_h by investigating the three activation curves of the extracted muscle synergies. The chosen a_h was used to yield $a = \alpha a_h$.

3) Designing a Reward Function by Applying the Proposed Approach: By using the identified a, we inferred the coefficients $c_{ax}, c_{ay}, c_p, c_v, c_\theta, c_\omega, c_p, c_v, c_\theta, c_\omega$ by solving the IOC problem with a numerical approach introduced in Sec. II-E. Consequently, we could define $r(s, a)$ explicitly; hence, this finalized the reward design process.

4) Training the RL Agent: With the designed reward function, a RL agent was trained under Unity ML-agent framework using the derived ball and flat-plate model. Proximal policy optimization (based on A3C architecture [32]) was employed as a learning algorithm and hyper-parameters were set empirically. The hyper-parameters settings are listed in APPENDIX.

For comparative analysis, RL agents were trained with various (but with typical characteristics) reward functions. The coefficients to define $r(s, a)$ were presented in Table III. We note that the coefficients for cheap, balanced, and expensive control were set by following conventional LQR sense, e.g., c_{ax}:c_p were set to 0.01:1 (cheap), 1:1 (balanced), and 100:1 (expensive). In contrast, the coefficients for the proposed approach (denoted by "Ours") were inferred by solving the IOC problem.

5) Evaluating the RL Agent: The comparative analysis for RL agents trained with $r(s, a)$ in Table III were performed. To all the four RL agents, a patience cube with 20 pseudo-random initial ball positions were given. The RL agents were evaluated by the following performance criteria: success rate, task-completion time, total control effort, trajectory length. All performance criteria were averaged out of 20 trials according to the initial positions.

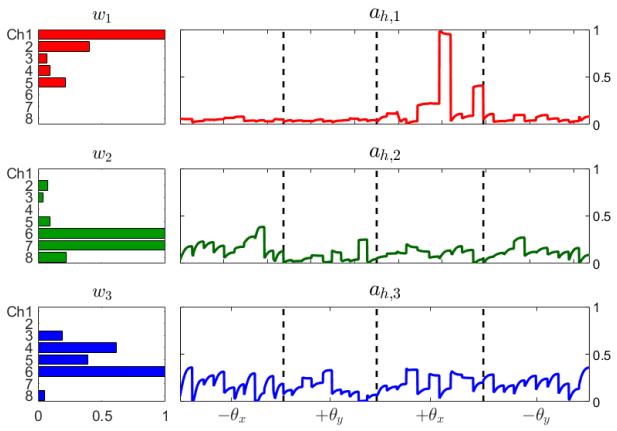

Fig. 6. The extracted muscle synergies and the corresponding activation curves.

Our source code and data set can be found at https://github.com/JanghyeonKimKR/IROS2023_Public.

IV. RESULTS

A. Muscle Synergies, Activation Curves, and Identified Control Input a

Fig. 6 shows the extracted muscle synergies and the corresponding activation curves. The left column of Fig. 6 presents i^{th} muscle synergy w_i in which the contributions of an individual muscle are presented as bar graphs. In the left column of Fig. 6, the activation curves are presented by value ranging $[0, 1]$ (the level of activation) along the horizontal axis (action types, $-\theta_x$: supination, $+\theta_y$: rotate upward, $+\theta_x$: pronation, $-\theta_y$: rotate downward and times).

By investigating the channels of contributing muscles in muscle synergies, we could identify that w_1 used ch#1 at ulna side as a major muscle. w_2 and w_3 commonly used ch#6 muscle; however, w_2 used the ulna side ch#7 simultaneously whereas w_3 used the radius side ch#4 and ch#5. Furthermore, observing the periodicity of the activation curves allowed us to tell a specific role for each muscle synergy. For the experiment introduced in Sec. III-B.2, the subject was instructed to perform periodic movement, e.g., pronation and supination. Therefore, we could identify that w_2 and w_3 were $a_{h,x}$ and $a_{h,y}$, which in turn, became $\alpha\varphi(a_{h,x}) =: a_x$ and $\alpha\varphi(a_{h,y}) =: a_y$, respectively. From the activation characteristic, we might say that w_1 might play a role of a gravity compensation as well as a grip force generating; however, further investigation is needed.

B. Designed Reward Function $r(s, a)$

For our proposed approach, indeed, the inferred coefficients values have already been shown at the bottom row in Table III, which is $c_{ax} = c_{ay} = 0.0201$, $c_p = 1$, $c_v = 0.0001$, $c_\theta = 1.2538$, and $c_\omega = 0.0004$. By substituting these values into

$$G(s, a) = \sum_{k=0}^{N-1} r(s_k, a_k) = \sum_{k=0}^{N-1} -\left[a_k^T R a_k + s_k^T Q s_k\right],$$

we could finally have an explicit form of $G(s, a)$.

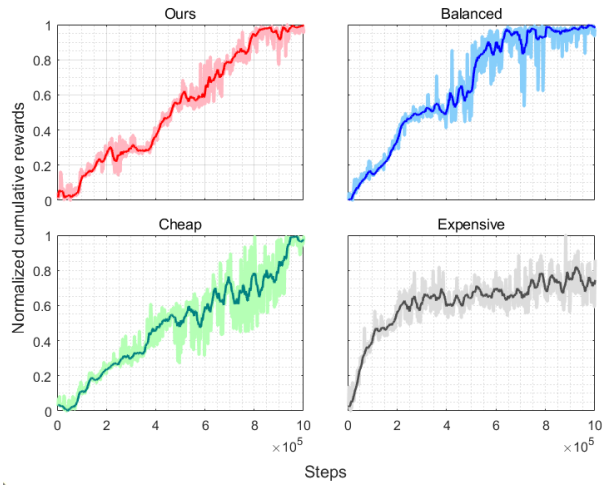

Fig. 7. Cumulative reward during the training according to the RL agents trained under different $r(s, a)$.

C. Performance Comparative Analysis for the Trained RL Agents

As aforementioned, four RL agents were trained, of which reward was set as in Table III, to perform comparative analysis versus our approach.

Fig. 7 shows a cumulative rewards value as training step being proceeded. From Fig. 7, we can see that the rank of convergence speed is as follows: expensive control, balanced control, ours, and cheap control (the higher rank is, the faster the RL agent converges during the training). Although, the expensive control showed the fastest convergence speed, it marked lower cumulative reward compared to ours. The balance control showed faster convergence speed over ours, but it was rather suffered by larger standard deviation during the training. The cheap control recorded slower convergence speed and larger standard deviation than those of ours. This result could be interpreted as our approach was able to generate human-like policy/strategy during the training (i.e. optimized but also somewhat heuristic).

Fig. 8 illustrates the examples of resulting (x, y) trajectories of the four RL agents with respect to three different initial positions. In this figure, the horizontal and the vertical axes are aligned to x- and y-axis, respectively. The green border represents the boundary of the patience cube plate. Our approach showed the best performance for the second and the third cases, but under-performed compared to the balanced control in the first case. The cheap control showed good performance overall; interestingly, it generated a jerky behavior in the third case. We can easily see that the expensive control hit the boundary wall and eventually failed for all three cases.

Table IV presents the performances of the RL agents which are averaged over 20 initial starting positions. Our approach marked the best performance in criteria of success rate and task-completion time in average, and it recorded the third and the second performance in terms of total control effort and travel length. Indeed, this outcome could be expected,

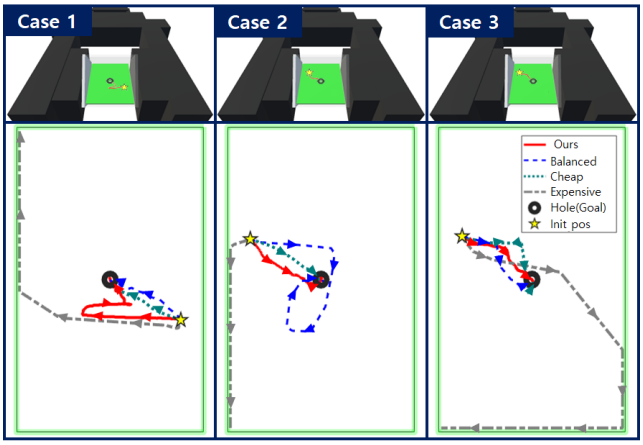

Fig. 8. The examples of resulting (x, y) trajectories performed by the 2-DoF robotic wrist-hand under the four RL agents with respect to three different initial positions.

TABLE IV
COMPARATIVE RESULT W.R.T. PERFORMANCE CRITERIA

Reward designed as/by	**Success rate** [%]	**Task Completion time** [sec]	**Control effort** $\sum$ [deg^2/sec]	**Travel length** [cm]
Cheap control	55	45.84	1039.50	24.14
Balanced control	50	44.21	584.92	**8.23**
Expensive control	0	N/A	**193.91**	22.36
Ours*	**95**	**33.74**	772.84	12.99

because the objective of our reward function design approach was to generate human-like behavior.

V. CONCLUSIONS

Throughout this paper, we proposed an approach to design a reward function which enables an RL agent to generate human-like policy/strategy. Our approach has a novelty in the problem of reward design in perspective of using both kinematic data and sEMG data obtained from human demonstration. The muscle synergy and inverse optimal control were utilized as key tools to implement our proposed approach. Comparative analysis results showed us that the trained RL agent under our approach could able to generate human-like policy/strategy as well as out-performance over some typical approaches.

The future work should be followed in the direction of identifying more elaborated relationship or mapping between a human effort and a control input in systemic level. We are envisioning to apply neural network-based approach as one of promising solutions. This study will be culminated to the design of an assistive robotic manipulator or prosthesis with individual-customization as well as high-dexterity for the activities of daily living.

APPENDIX

The main patience cube ML-agents hyper-parameter settings for training a proximal policy optimization (PPO) network are as follows:

- Learning rate: 1.0e-3
- Learning rate schedule: linear
- Number of hidden layers : 3
- Number of hidden units in each layer: 128
- Batch size : 512
- Buffer size : 4096
- Normalize : false
- Beta : 5.0e-3
- Epsilon : 0.2
- Lambda : 0.95

For more details about PPO hyper-parameter setting, see [33].

REFERENCES

[1] V. Kumar, E. Todorov, and S. Levine, "Optimal control with learned local models: Application to dexterous manipulation," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 378–383, 2016.

[2] T. Chen, J. Xu, and P. Agrawal, "A system for general in-hand object re-orientation," in *Conference on Robot Learning*, pp. 297–307, PMLR, 2022.

[3] H. Zhu, A. Gupta, A. Rajeswaran, S. Levine, and V. Kumar, "Dexterous manipulation with deep reinforcement learning: Efficient, general, and low-cost," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 3651–3657, IEEE, 2019.

[4] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, *et al.*, "Learning dexterous in-hand manipulation," *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020.

[5] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine, "Learning complex dexterous manipulation with deep reinforcement learning and demonstrations," *arXiv preprint arXiv:1709.10087*, 2017.

[6] I. Akkaya, M. Andrychowicz, M. Chociej, M. Litwin, B. McGrew, A. Petron, A. Paino, M. Plappert, G. Powell, R. Ribas, *et al.*, "Solving rubik's cube with a robot hand," *arXiv preprint arXiv:1910.07113*, 2019.

[7] S. Levine, N. Wagener, and P. Abbeel, "Learning contact-rich manipulation skills with guided policy search (2015)," *arXiv preprint arXiv:1501.05611*, 2015.

[8] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1334–1373, 2016.

[9] Y. Duan, X. Chen, R. Houthooft, J. Schulman, and P. Abbeel, "Benchmarking deep reinforcement learning for continuous control," in *International conference on machine learning*, pp. 1329–1338, PMLR, 2016.

[10] E. Todorov and M. I. Jordan, "Optimal feedback control as a theory of motor coordination," *Nature neuroscience*, vol. 5, no. 11, pp. 1226–1235, 2002.

[11] S. Levine and V. Koltun, "Continuous inverse optimal control with locally optimal examples," *arXiv preprint arXiv:1206.4617*, 2012.

[12] K. Dvijotham and E. Todorov, "Inverse optimal control with linearly-solvable mdps," in *Proceedings of the 27th International conference on machine learning (ICML-10)*, pp. 335–342, 2010.

[13] A. Keshavarz, Y. Wang, and S. Boyd, "Imputing a convex objective function," in *2011 IEEE international symposium on intelligent control*, pp. 613–619, IEEE, 2011.

[14] A.-S. Puydupin-Jamin, M. Johnson, and T. Bretl, "A convex approach to inverse optimal control and its application to modeling human locomotion," in *2012 IEEE International Conference on Robotics and Automation*, pp. 531–536, IEEE, 2012.

[15] E. Todorov, "Optimality principles in sensorimotor control," *Nature neuroscience*, vol. 7, no. 9, pp. 907–915, 2004.

[16] R. Wen, K. Yuan, Q. Wang, S. Heng, and Z. Li, "Force-guided high-precision grasping control of fragile and deformable objects using semg-based force prediction," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2762–2769, 2020.

[17] J. M. Fajardo, O. Gomez, and F. Prieto, "Emg hand gesture classification using handcrafted and deep features," *Biomedical Signal Processing and Control*, vol. 63, p. 102210, 2021.

[18] A. Furui, S. Eto, K. Nakagaki, K. Shimada, G. Nakamura, A. Masuda, T. Chin, and T. Tsuji, "A myoelectric prosthetic hand with muscle synergy–based motion determination and impedance model–based biomimetic control," *Science Robotics*, vol. 4, no. 31, p. eaaw6339, 2019.

[19] J.-H. Yoo, H.-J. Jung, J.-H. Kim, D.-H. Sim, and H.-U. Yoon, "Solving a simple geduldspiele cube with a robotic gripper via sim-to-real transfer," *Applied Sciences*, vol. 12, no. 19, 2022.

[20] N. Bernstein, "The co-ordination and regulation of movements," *The co-ordination and regulation of movements*, 1966.

[21] L. H. Ting and J. M. Macpherson, "A limited set of muscle synergies for force control during a postural task," *Journal of neurophysiology*, vol. 93, no. 1, pp. 609–613, 2005.

[22] J. L. Allen, H. D. Carey, L. H. Ting, and A. Sawers, "Generalization of motor module recruitment across standing reactive balance and walking is associated with beam walking performance in young adults," *Gait & posture*, vol. 82, pp. 242–247, 2020.

[23] D. J. Berger and A. d'Avella, "Effective force control by muscle synergies," *Frontiers in computational neuroscience*, vol. 8, 2014.

[24] J. M. Inouye and F. J. Valero-Cuevas, "Muscle synergies heavily influence the neural control of arm endpoint stiffness and energy consumption," *PLoS computational biology*, vol. 12, no. 2, p. e1004737, 2016.

[25] V. C. Cheung, A. d'Avella, M. C. Tresch, and E. Bizzi, "Central and sensory contributions to the activation and organization of muscle synergies during natural motor behaviors," *Journal of Neuroscience*, vol. 25, no. 27, pp. 6419–6434, 2005.

[26] A. d'Avella and E. Bizzi, "Shared and specific muscle synergies in natural motor behaviors," *Proceedings of the national academy of sciences*, vol. 102, no. 8, pp. 3076–3081, 2005.

[27] S. A. Chvatal and L. H. Ting, "Common muscle synergies for balance and walking," *Frontiers in computational neuroscience*, vol. 7, 2013.

[28] A. Sawers, Y.-C. Pai, T. Bhatt, and L. H. Ting, "Neuromuscular responses differ between slip-induced falls and recoveries in older adults," *Journal of neurophysiology*, vol. 117, no. 2, pp. 509–522, 2017.

[29] G. Dominijanni, S. Shokur, G. Salvietti, S. Buehler, E. Palmerini, S. Rossi, F. De Vignemont, A. d'Avella, T. R. Makin, D. Prattichizzo, *et al.*, "The neural resource allocation problem when enhancing human bodies with extra robotic limbs," *Nature Machine Intelligence*, vol. 3, no. 10, pp. 850–860, 2021.

[30] D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Advances in neural information processing systems*, vol. 13, 2000.

[31] A. d'Avella and F. Lacquaniti, "Control of reaching movements by muscle synergy combinations," *Frontiers in computational neuroscience*, vol. 7, 2013.

[32] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*, pp. 1928–1937, PMLR, 2016.

[33] A. Majumder, *Deep Reinforcement Learning in Unity: With Unity ML Toolkit*. Apress Berkeley, CA, 2021.