ASME Journal of Mechanisms and Robotics
Online journal at:
https://asmedigitalcollection.asme.org/mechanismsrobotics

Wenda Xu
Robotics and Mechatronics Laboratory,
Department of Mechanical Engineering,
Virginia Tech,
Blacksburg, VA 24061
e-mail: wenda@vt.edu

Yunfei Guo
Robotics and Mechatronics Laboratory,
Department of Electrical and Computer Engineering,
Virginia Tech,
Blacksburg, VA 24061
e-mail: yunfei96@vt.edu

Yujiong Liu
Robotics and Mechatronics Laboratory,
Department of Mechanical Engineering,
Virginia Tech,
Blacksburg, VA 24061
e-mail: yjliu@vt.edu

Pinhas Ben-Tzvi[1]
Robotics and Mechatronics Laboratory,
Department of Mechanical Engineering,
Virginia Tech,
Blacksburg, VA 24061
e-mail: bentzvi@vt.edu

Development of a Novel Compact Robotic Exoskeleton Glove With Reinforcement Learning Control

This paper presents the design, optimization, control, and experimental evaluation of a novel compact exoskeleton glove aiming to assist patients with brachial plexus injuries in grasping daily used objects. The finger mechanism is based on a rigid coupling hybrid mechanism concept, which utilizes a serially connected rack-and-pinion mechanism and an offset slider-crank mechanism to couple the motions of different finger joints. The glove dimensions are synthesized based on the natural grasping motion of human hands. To better control the glove and enhance the grasping capabilities, a simulation environment was developed and reinforcement learning techniques were applied. This learning-based control trained an agent to perform different grasp types with appropriate force. The trained agent was then applied in real-world experiments with the developed exoskeleton glove. The results validated the effectiveness of the mechanical design and the real-time self-adjustable control policy, which demonstrated the glove's functionality and capability to grasp various objects relevant to activities of daily living. [DOI: 10.1115/1.4064283]

Keywords: artificial intelligence-enhanced mechanical systems, grasping, fixturing, multi-finger hands, mechanism synthesis and analysis, mechanisms and robots, rehabilitation mechanisms and robots, wearable robots, prosthetics/orthotics, exoskeletons

1 Introduction

Brachial plexus injury (BPI) is a severe peripheral nerve injury affecting upper limbs and causing pain, weakness, and numbness in the arm and hand [1]. It is mostly caused by vehicle accidents, athletic injuries, and gunshot wounds. Patients who suffer from BPI will loss feeling and control ability in shoulder, arm, and hand. Previous studies have proven that the surgical options can be used to recover the functionalities of shoulder and arm. But the improvement on hand is marginal due to its complexity [2]. These limitations in human hand functionality severely impact patients' daily life and increase the assistance requirements for activities of daily living (ADLs).

Numerous research groups are dedicated to developing assistive devices to restore the lost functions and mobility in individuals with various neuromuscular disorders. Advanced technologies, such as exoskeleton devices, have paved the way for individuals with disabilities or diseases to a more independent live through the use of autonomous devices [3]. Considering the requirements of human–robot interaction, the design requirements for hand exoskeleton devices, specifically tailored for ADLs, can be summarized as follows [4]. First, the exoskeleton device should feature a compact design to minimize collisions with the environment. This compactness also contributes to a lightweight and comfortable structure. Second, the mechanism should be simple enough to enhance the reliability and safety while reducing the size and cost of the exoskeleton devices. Finally, the ergonomics of the human hand exoskeleton, including the remote center of motion (RCM) and human motion imitation, are crucial for increasing comfort and providing natural hand movements.

Typically, the hand exoskeleton mechanisms can be categorized into three types based on their actuation methods: pneumatic/hydraulic mechanism (mostly used for soft robotics [5–8]), Tendon/Bowden-driven mechanism [9,10], and linkage-driven mechanism [11–14]. Soft robotic gloves are effective in providing compliance, enhancing safety by absorbing impacts, and offering good wearability. However, they usually require thick inflatable segments attached to each finger for force generation. Moreover, designing precise trajectories is challenging due to the poor repeatability of elastic element deformation. The additional air compressor package also reduces the device portability. Tendon/Bowden-driven mechanisms are suitable for non-portable hand exoskeletons as they allow motors to be placed remotely. However, motion transmission in these mechanisms is prone to introduce friction, backlash, and to have a limited maximum force due to sheathes exerting forces during movement. In comparison, linkage-driven methods provide a simple and direct approach to transfer force and motion from the actuators to the fingertips. The motors can

[1]Corresponding author.

Contributed by Mechanisms and Robotics Committee of ASME for publication in the Journal of Mechanisms and Robotics. Manuscript received August 16, 2023; final manuscript received December 11, 2023; published online January 12, 2024. Assoc. Editor: Joo H. Kim.

be integrated into the exoskeleton to enhance portability. However, a delicate design is necessary to minimize its size while considering ergonomics.

Due to the various tasks involved in grasping different objects with different shapes and weights, simple force control algorithms may not be sufficient to successfully complete these tasks. Elaborate grasping force control may be necessary for precise manipulation in real-world interactions, which relies on robust sensing capabilities. However, according to a survey, only less than half of the surveyed studies demonstrated a force control implementation in hand exoskeleton systems [15]. This indicates the need for further development and research in this area to improve the effectiveness and applicability of force control algorithms in exoskeleton devices.

To enhance the overall performance of robotic gloves in terms of wearability, comfort, and portability, we developed a novel exoskeleton glove in this paper. This design addresses the limitations of existing wearable devices while incorporating the general requirements of hand exoskeleton devices. Figure 1 illustrates a proof-of-concept prototype of this proposed exoskeleton glove.

In the new glove design, each finger linkage mechanism is actuated by a customized linear actuator. The motion of each finger exoskeleton is coupled by a rigid coupling hybrid mechanism (RCHM) which helps to reduce the number of degrees-of-freedom (DOF) and improve the overall functionality. To ensure a natural and comfortable user experience, the glove mechanism dimensions are carefully synthesized to imitate the natural motion of human hands. This approach improves wearability and comfortability, allowing users to perform various ADLs easily. Furthermore, the compact and low-profile design of the exoskeleton glove enhances its portability, allowing users to wear it comfortably in different environments. To enhance the glove's versatility, a reinforcement learning-based force control policy is implemented. This enables the glove to effectively solve different grasping tasks by autonomously adjusting the applied forces.

The main contributions of this work can be summarized as follows. First, we have expanded the novel low-profile mechanism [16] that exhibits mechanical stability and robustness, initially developed for two fingers, to encompass the entire hand. Second, we successfully developed a proof-of-concept prototype based on the proposed mechanism and the synthesized results. Third, we developed a simulation environment for the new glove using SIMSCAPE and trained a reinforcement learning agent using a deep deterministic policy gradient (DDPG) to generate a force control policy which is capable of handling different grasp types and objects. Finally, we conducted comprehensive experiments to evaluate the effectiveness of our design and proposed control policy by calibrating force for each actuator and bridging the gap between the simulation environment and the real-world.

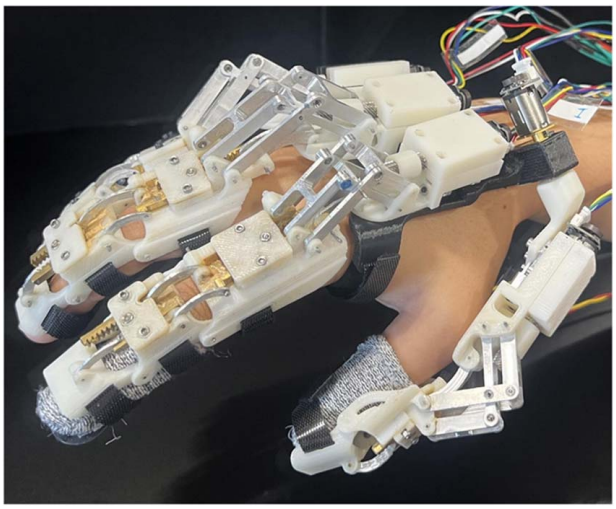

Fig. 1 A proof-of-concept prototype of the new exoskeleton glove

Compared to other hand exoskeletons, our new exoskeleton glove exhibits several unique characteristics: (1) it utilizes a linkage-driven mechanism, which ensures mechanical stability and robustness, (2) the finger mechanism is based on the RCHM concept, resulting in a compact and low-profile design, (3) the exoskeleton glove is synthesized to closely imitate the actual motion of the human hand, enabling better grasping performance for ADLs for patients, and (4) the glove implements a reinforcement learning-based force control policy that generates appropriate forces in real-time for different grasp tasks.

The remainder of this paper is organized as follows. Section 2 presents the mechanism and mechanical design of the exoskeleton glove, including the kinematics model and the design variables. Section 3 discusses the electrical design and the low-level control method. Section 4 provides the simulation environment details and an overview of the reinforcement learning method that is used to generate the appropriate force control policy. Section 5 details the experiments conducted on healthy subjects. Finally, Sec. 6 concludes the paper, summarizes the findings, and discusses potential future directions.

2 Mechanism and Mechanical Design

This section explores the design principles of the finger mechanism and outlines the process of optimizing the motion trajectories for each finger. Additionally, we will present the final configurations of the prototype that were implemented. It is worth noting that the total weight of the entire glove system, excluding electronics and batteries, amounts to 304 g.

2.1 Mechanism Design. The design of the proposed exoskeleton glove incorporates the concept of coupling motion between different phalanges within each finger during grasping. Building upon previous research findings [17–19], which proves that reducing the number of DOFs in the hand is sufficient for performing most activities of daily living, three DOFs of each finger (excluding abduction–adduction) is simplified into a single-DOF. By doing so, we reduce the number of required actuators, resulting in a more compact and lightweight design.

The simplified finger mechanism of the exoskeleton glove is based on the one DOF case of the rigid coupling hybrid mechanism concept [20]. The motion transmission is achieved by coupling the motion of each link with its adjacent link, allowing the motion of the current link to be driven by its neighboring links rather than the actuator itself. To implement the RCHM, two fundamental components are required: a parallel mechanism (PM) that determines the basic mobility of the mechanism, and a rigid coupling mechanism that couples the motions of the adjacent PMs. The rigid coupling mechanism can be realized in various forms, such as a rack-and-pinion mechanism or a four-bar mechanism, depending on the specific design requirements.

Each finger exoskeleton comprises three links according to human hand anatomy: a distal link, a middle link, and a proximal link. Additionally, three relative joints are included: a distal interphalangeal joint (DIP), a proximal interphalangeal joint (PIP), and a metacarpophalangeal joint (MCP). The thumb digit exoskeleton follows a similar configuration, but lack of the middle link and DIP joint.

To ensure a proper fit on the patient's hand, each link of the exoskeleton finger is aligned with the corresponding phalanx, and each joint corresponds to the respective joint on the patient's hand. The finger mechanism utilizes a combination of offset slider-crank mechanisms and rack-and-pinion mechanisms to connect different links. Referring to Fig. 2, the principles of the finger mechanism are as follows:

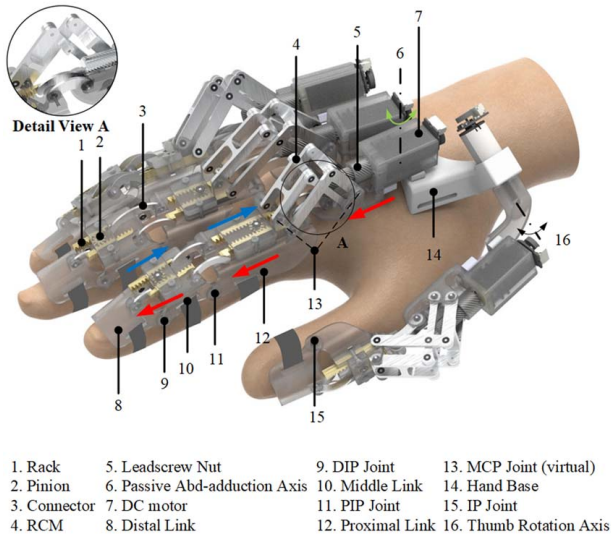

Fig. 2 The overview of the proposed mechanism design. Two different mechanisms on the exoskeleton finger are added: a rack-and-pinion mechanism for motion reversing and an offset slider-crank mechanism for motion conversion. The arrow pointed to the left represents the driving chain which drives the next connected linkage. The arrow pointed to the right represents the measuring chain which measures the movement of the driving chain.

(1) A customized linear actuator consists of the motor and leadscrew is actuated by the DC motor mounted at the end. The leadscrew nut is connected to the proximal link through a connector. This connection forms the first offset slider-crank mechanism around the MCP joint, which converts the linear motion generated by the linear actuator into a rotary motion of the MCP joint. The rotary motion of the MCP joint drives the proximal link accordingly.
(2) Furthermore, the housing of the linear actuator is connected to the sliding rack using another connector. This connection forms the second offset slider-crank mechanism around the MCP joint, which converts the rotary motion of the MCP joint into the linear motion of the rack. Thus, the motion originating from the linear actuator is transmitted to the linear motion of the rack on the proximal link, but with a different direction due to the rotation of the MCP joint.
(3) After the transmission of motion through the second offset slider-crank mechanism, the pinion-rack mechanism is employed to reverse the direction of motion and establish a connection between the second offset slider-crank mechanism around the MCP joint and the first offset slider-crank mechanism around the PIP joint. By such way, the motion of the MCP joint and the motion of PIP joint are coupled.
(4) The same process is repeated to transmit motion from the PIP joint to the DIP joint. Through these successive connections of the pinion-rack mechanisms and offset slider-crank mechanism, the motion of all three different joints (MCP, PIP, and DIP) on the same exoskeleton finger is coupled.

It is important to note that the MCP joint on the exoskeleton finger is virtual and replaced by a RCM mechanism to avoid interference with the patient's hand.

The first offset slider-crank mechanism, represented by the red arrow in Fig. 2, is called the "driving" mechanism. It generates forward motion and drives the motion of the MCP joint in the finger mechanism. Conversely, the second offset slider-crank mechanism, represented by the blue arrow, is called the "measuring" mechanism. It generates backward motion and its motion depends on the driving mechanism, effectively measuring the output motion of the MCP joint. The rotation angle of the MCP joint, produced by the driving mechanism, is measured by the measuring mechanism and presented as the displacement of the rack.

A similar arrangement exists around the PIP joint, where another pair of driving and measuring mechanisms are present. The rack-and-pinion mechanism connects the measuring mechanism of the MCP joint to the driving mechanism of the PIP joint, resulting in the coupling motion between the MCP joint and the PIP joint. The achievement of proposed sequential transmission of motion from the linear actuator to the end effector (distal link), enables coordinated and synchronized finger joints' movements in the exoskeleton glove.

It is worth noting that there is no measuring mechanism around the DIP joint since it is already connected to the last link, and its motion is directly coupled to the previous joint.

Additionally, a passive abduction and adduction mechanism with mechanical limits is incorporated. This mechanism introduces a revolute joint at the base of the actuator housing, connecting the finger linkage and the hand base. It enables comfortable movement for the patient while ensuring appropriate alignment.

This unique design allows us to make the exoskeleton finger mechanism as thin as possible and attach the whole mechanism to the top of each finger to avoid interference between the exoskeleton glove and the human hand, improving the wearability, comfortability, and portability of the glove.

2.2 Kinematic Analysis. Hand kinematics includes the four-finger kinematic chains and the thumb kinematic chains. Since the motion of different finger joints has similar coupling motion, their kinematics are summarized into a single formulation. The spatial kinematic chain from base to thumb Metacarpal is treated separately. One finger and thumb are used to simplify the model for illustration, as shown in Fig. 3.

The proposed exoskeleton glove incorporates five distinct kinematic chains to accommodate different digits. Among them, the four fingers exhibit similar coupled planar motion, allowing their kinematics to be represented by a unified formulation. On the other hand, the kinematic chain of the thumb takes both the coupled planar motion and the spatial motion around the Carpo–Metacarpa (CMC) joint into account. By considering these variations, the exoskeleton glove achieves comprehensive control over the hand movements, accommodating the unique kinematics of each digit [16,21].

For the finger linkage kinematic analysis, the forward kinematic chain of the current finger c is given by Eq. (1)

$$p_{x,c} + ip_{y,c} = l_{p,c}e^{i\theta_{mcp,c}} + l_{m,c}e^{i(\theta_{mcp,c}+\theta_{pip,c})} + l_{d,c}e^{i(\theta_{mcp,c}+\theta_{pip,c}+\theta_{dip,c})} \tag{1}$$

where $c \in \{index, middle, ring, little\}$ represents the current finger, $p_{x,c}$, $p_{y,c}$ are the x and y coordinates of the exoskeleton fingertip position in the finger motion plane. $\theta_{mcp,c}$, $\theta_{dip,c}$, and $\theta_{pip,c}$ are the rotating angles of the MCP joint, the PIP joint, and the DIP joint on the current exoskeleton finger, respectively. $l_{p,c}$, $l_{m,c}$, and $l_{d,c}$ are the lengths of the proximal link, the middle link, and the distal link on the current exoskeleton finger, respectively.

Based on the mechanism design that is discussed in Sec. 2, the following relationship can be obtained:

$$A_{c,jnt,i} = \sqrt{B^2_{c,jnt,i} - C^2_{c,jnt,i}} + D_{c,jnt,i} - E_{c,jnt,i} \tag{2}$$

where

$$A_{c,jnt,i} = (l_{c,jnt,i} - d_{c,jnt})\cos\alpha_{c,jnt,i} \tag{3}$$

$$B_{c,jnt,i} = l^2_{c,jnt,i}\cos\alpha_{c,jnt,i} \tag{4}$$

$$C_{c,jnt,i} = h^2_{c,jnt}(\cos\alpha_{c,jnt,i} - \cos(\alpha_{c,jnt,i} + \theta_{c,jnt})) \tag{5}$$

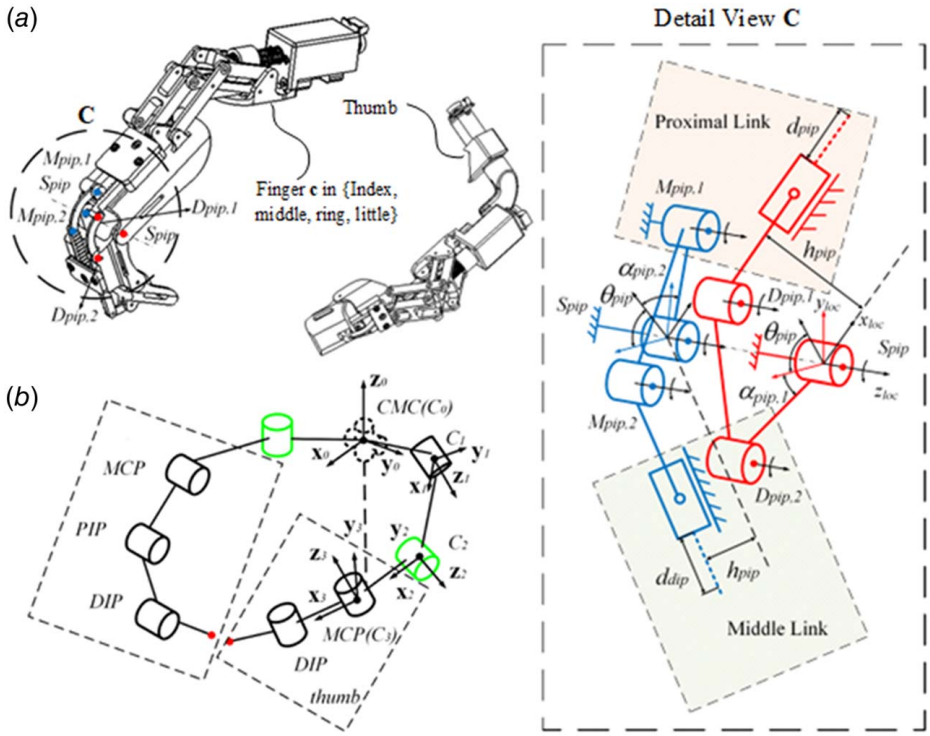

Fig. 3 Kinematic model of the exoskeleton glove mechanism with c-th finger and thumb. Detail view C shows the kinematic model around the PIP joint on the finger c for illustration. (*a*) CAD model of the c-th finger and the thumb and (*b*) kinematic chain representation of the c-th finger and the thumb.

$$D_{c,jnt,i} = h_{c,jnt} \sin \alpha_{c,jnt,i} \tag{6}$$

$$E_{c,jnt,i} = h_{c,jnt} \cos (\alpha_{c,jnt,i} + \theta_{c,jnt}) \tag{7}$$

and $jnt \in \{mcp, pip, dip\}$, $i \in \{1, 2\}$. Subscript $i = 1$ refers to the driving chain and subscript $i = 2$ refers to the measuring chain. l is the connector length, h is the distance between the joint and the middle plane of the rack, and α is the angle of the crank.

The kinematics of the human thumb involve three phalanges: the proximal phalanx, the distal phalanx, and the metacarpal phalanx. The motion of the proximal and distal phalanges can be captured using the finger linkage kinematics described earlier, as they exhibit planar motion.

However, the metacarpal phalanx is connected to the Carpus through the CMC joint, which provides two DOF. To replicate the functionalities of the CMC joint in the exoskeleton, it is divided into two independent revolute joints, denoted as C_1 and C_2 [16] as shown in Fig. 3. It is important to note that the second revolute joint (C_2) is designed to be passive, ensuring consistency with other finger exoskeletons.

The CMC joint (C_0) of the thumb is chosen as the origin of the global coordinate system. Each link frame, denoted as $\sum C_j$ where $0 \le j \le 3$, is defined at the axis of joint j, with $\mathbf{z}_j$ coinciding with the joint axis and $\mathbf{x}_j$ pointing toward the next joint. The forward kinematics from the CMC joint on the hand to the MCP joint on the thumb can be expressed as follows:

$$\mathbf{R}_{C_j} = \mathbf{R}_{C_{j-1}} \mathbf{R}_z(\beta_{C_j,z}) \mathbf{R}_y(\beta_{C_j,y}) \mathbf{R}_x(\beta_{C_j,x}) \mathbf{R}_z(\theta_{C_j}) \tag{8}$$

$$\mathbf{p}_{C_j} = \mathbf{R}_{C_{j-1}} \mathbf{p}_{C_{j-1},C_j} + \mathbf{p}_{C_{j-1}} \tag{9}$$

where $\mathbf{R}_{C_j}$ denotes the orientation of the frame $\sum C_j$, $\mathbf{R}_x(\cdot)$, $\mathbf{R}_y(\cdot)$, and $\mathbf{R}_z(\cdot)$ denote the principle rotation matrix functions with respect to the corresponding axis, $\boldsymbol{\beta}_{C_j} = [\beta_{C_j,x}\ \beta_{C_j,y}\ \beta_{C_j,z}]^T$ denotes the vector of rotation angles with respect to the corresponding axis, θ_{C_j} denotes the rotation angle of the joint C_j, $\mathbf{p}_{C_j}$ denotes the C_j position, $\mathbf{p}_{C_{j-1},C_j}$ denotes the vector from C_{j-1} to C_j. For the initial conditions, $\mathbf{R}_{C_0}$ is the identity matrix, θ_{C_0} is zero, $\mathbf{p}_{C_0}$ and $\boldsymbol{\beta}_{C_0}$ are zero vectors.

2.3 Kinematic Synthesis. The finger mechanism requires proper alignment of joint axis on the exoskeleton and the joint axis on the human finger. However, due to the significant size and shape variations among individuals, an optimization process is conducted based on the finger dimensions provided by the author, as indicated in Table 1. It is worth mentioning that the thumb does not have a middle link, and the parameter l_c for four fingers are ignored in the optimization process, as it has minimal impact on the finger motion.

The design of the exoskeleton glove aims to replicate the natural grasping motion of the human hand. To achieve this, the design variables of the exoskeleton glove are optimized based on the UNIPI dataset [22], which provides joint trajectories of the fingers during grasping motions. Specifically, the finger joint data from the No. 1 subject while grasping the No. 8 object (grasping the tape with five digits) are selected, as they contain long trajectories of joint motion, thus expanding the exoskeleton glove's workspace.

Since the selected trajectories cover both grasping and releasing motions and contain thousands of data points, preprocessing is

Table 1 Finger dimensions

Name	l_c (mm)	l_p (mm)	l_m (mm)	l_d (mm)
Thumb	52.0	36.0	—	34.0
Index	—	50.0	27.2	26.4
Middle	—	57.0	35.4	24.2
Ring	—	55.1	33.0	24.6
Little	—	44.7	25.4	24.0

Table 2 Optimized design variables

Index finger				Middle finger			
Variable	Value	Variable	Value	Variable	Value	Variable	Value
$l_{mcp,1}$	13.0 mm	$l_{mcp,2}$	13.9 mm	$l_{mcp,1}$	20.0 mm	$l_{mcp,2}$	19.3 mm
$l_{pip,1}$	15.0 mm	$l_{pip,2}$	15.6 mm	$l_{pip,1}$	14.3 mm	$l_{pip,2}$	21.7 mm
$l_{dip,1}$	8.2 mm	h_{mcp}	16.7 mm	$l_{dip,1}$	12.0 mm	h_{mcp}	14.3 mm
h_{pip}	10.0 mm	h_{dip}	10.0 mm	h_{pip}	10.0 mm	h_{dip}	10.0 mm
$\alpha_{mcp,1}$	17.1 deg	$\alpha_{mcp,2}$	22.3 deg	$\alpha_{mcp,1}$	17.8 deg	$\alpha_{mcp,2}$	17.2 deg
$\alpha_{pip,1}$	24.6 deg	$\alpha_{pip,2}$	30.0 deg	$\alpha_{pip,1}$	29.4 deg	$\alpha_{pip,2}$	30.0 deg
$\alpha_{dip,1}$	45.0 deg			$\alpha_{dip,1}$	47.1 deg		
Ring finger				Little finger			
$l_{mcp,1}$	17.2 mm	$l_{mcp,2}$	17.8 mm	$l_{mcp,1}$	13.4 mm	$l_{mcp,2}$	13.1 mm
$l_{pip,1}$	12.2 mm	$l_{pip,2}$	19.5 mm	$l_{pip,1}$	12.2 mm	$l_{pip,2}$	13.0 mm
$l_{dip,1}$	12.2 mm	h_{mcp}	16.0 mm	$l_{dip,1}$	11.2 mm	h_{mcp}	16.0 mm
h_{pip}	10.0 mm	h_{dip}	10.0 mm	h_{pip}	10.0 mm	h_{dip}	10.0 mm
$\alpha_{mcp,1}$	18.0 deg	$\alpha_{mcp,2}$	19.7 deg	$\alpha_{mcp,1}$	17.8 deg	$\alpha_{mcp,2}$	17.2 deg
$\alpha_{pip,1}$	25.8 deg	$\alpha_{pip,2}$	30.0 deg	$\alpha_{pip,1}$	22.1 deg	$\alpha_{pip,2}$	30.0 deg
$\alpha_{dip,1}$	45.0 deg			$\alpha_{dip,1}$	45.0 deg		
Thumb							
$l_{mcp,1}$	14.9 mm	$l_{mcp,2}$	20 mm	$l_{dip,1}$	15.9 mm	h_{mcp}	16.1 mm
h_{dip}	10.0 mm	$\alpha_{mcp,1}$	17.2 deg	$\alpha_{mcp,2}$	10.0 deg	$\alpha_{dip,1}$	30.0 deg
$p_{C_0,C_1,x}$	−14.8 mm	$\beta_{C_1,x}$	−154.5 deg	$p_{C_0,C_1,y}$	10.1 mm	$\beta_{C_1,y}$	−170.2 deg
$p_{C_0,C_1,z}$	16.3 mm	$\beta_{C_1,z}$	−128.8 deg	$p_{C_1,C_2,x}$	2.9 mm	$\beta_{C_1,x}$	115.7 deg
$p_{C_1,C_2,y}$	−24.0 mm	$\beta_{C_1,y}$	170.4 deg	$p_{C_1,C_2,z}$	−18.6 mm	$\beta_{C_1,z}$	38.6 deg
l_{C_2,C_3}	40.0 mm						

performed to extract only the grasping motion. From the grasping data, four equally distributed positions along the trajectory are chosen as landmarks. Each landmark is defined by 15 joint angles (three distinct joints on each digit), which characterize the hand's posture.

The objective function is formulated as the weighted sum of the root mean squared error (RMSE) between the joint angles of the landmarks and the corresponding joint angles of the exoskeleton fingers. The RMSE is calculated for each joint angle, resulting in a vector $\mathbf{e} \in \mathbb{R}^{60\times1}$. The weights $\mathbf{w}_k \in \mathbb{R}^{1\times15}$ are assigned to each joint angle of the kth landmark, and they reflect the importance of each landmark in the optimization. The objective function Z is defined as

$$Z = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3 \quad \mathbf{w}_4]\mathbf{e} \tag{10}$$

The selection of weights for the MCP, PIP, and DIP joints considers the different range of motion of each joint and its relationship with the adjacent joints. This allows the optimization to consider the relative importance of each joint angle in achieving a more accurate replication of the human motion trajectory [23]

$$0\,\text{deg} \le \theta_{mcp} \le 90\,\text{deg} \tag{11}$$

$$0\,\text{deg} \le \theta_{pip} \le 110\,\text{deg} \tag{12}$$

$$0\,\text{deg} \le \theta_{dip} \le 90\,\text{deg} \tag{13}$$

For the finger linkage mechanism, additional nonlinear inequality constraints are introduced to incorporate the rack-and-pinion mechanisms. These constraints ensure that the motion of the rack and the rotation of the pinion are properly synchronized. The specific form of these constraints depends on the geometric characteristics and design parameters of the rack-and-pinion mechanism used in the exoskeleton glove.

$$2d_{pip} \le l_p - h_{mcp}\cos(\alpha_{mcp,1}) - h_{pip}\cos(\alpha_{pip,2}) \tag{14}$$

$$2d_{dip} \le l_m - h_{pip}\cos(\alpha_{pip,1}) - h_{dip}\cos(\alpha_{dip,2}) \tag{15}$$

$$h_{pip} \le h_{mcp} \tag{16}$$

$$h_{dip} \le h_{pip} \tag{17}$$

The five digits are optimized together to achieve a better performance. The optimization problem is solved by the MATLAB function fmincon with the "interior-point" algorithm. The optimized design variables are collected in Table 2.

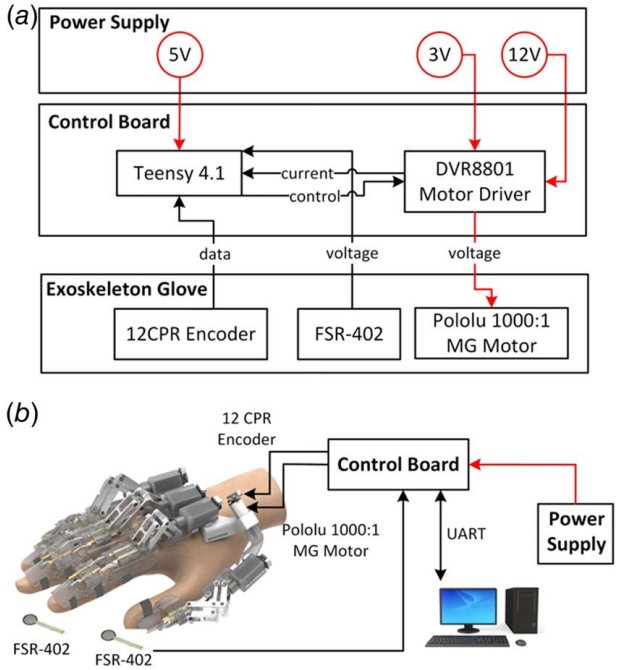

Fig. 4 (*a*) Electronics block diagram and (*b*) rendered exoskeleton glove with electronics

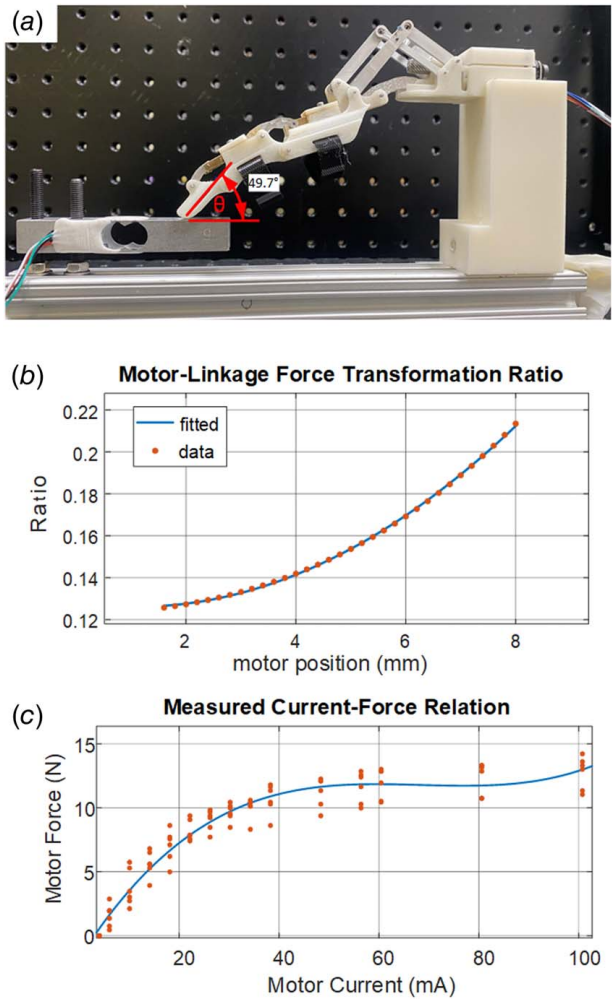

Fig. 5 Current–force conversion experiment setup and result: (*a*) current–force conversion experiment setup, (*b*) actuator to fingertip linkage force transformation ratio as a function of actuator position, and (*c*) actuator output force as a function of motor current consumption

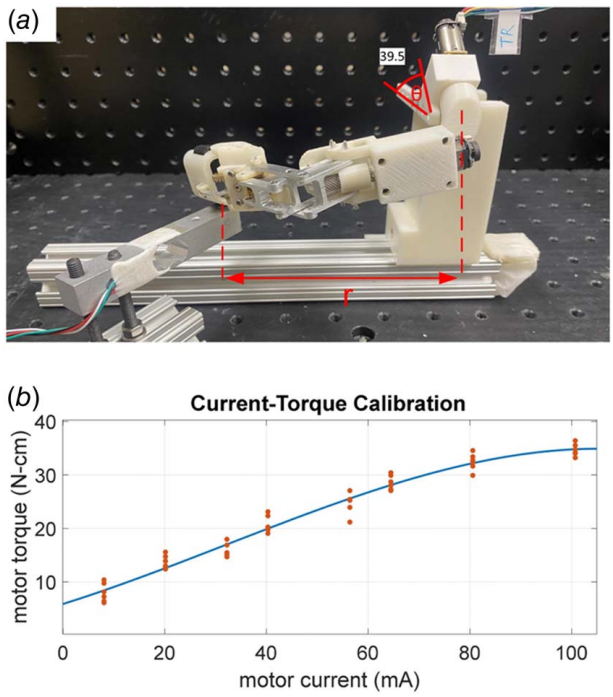

Fig. 6 Current to torque calibration for the rotatory actuator: (*a*) current–torque conversion experiment setup and (*b*) actuator output torque as a function of motor current consumption

3 Electronics and Control

Figure 4 illustrates the electronics block diagram of the system. The electronics system is divided into three parts.

The first part comprises power supply units that provide 3 V, 5 V, and 12 V voltage output.

The second part is the control board, consists of a Teensy 4.1 microcontroller and seven DVR8801 motor drivers. The onboard Teensy 4.1 microcontroller handles sensor readings and performs various low-level control tasks, such as motor current control, motor force control, and motor proportional–integral–derivative (PID) position control. To minimize sensor reading latency and enable parallel computing, a real-time system is employed. The microcontroller communicates with a desktop computer to perform deep reinforcement learning (DRL).

Third, the exoskeleton glove uses six dual-shaft 12V 1000:1 Pololu metal gear motors to produce sufficient power and torque to generate motion in the proposed exoskeleton glove (one for each finger and one for the thumb thenar). Each motor is paired with a DVR-8801 motor driver and 12-CPR magnetic encoders. The electrical current consumption of the motor can be measured using the DVR-8801 motor driver. We use a fifth order low-pass Butterworth filter with a 2 Hz cutoff frequency to filter the measured current. Two Interlink FSR-402 force sensitive resistors (FSRs) are placed on wearable finger sleeves on the thumb and index finger. Both FSRs are connected to a 3.3 V voltage source with a 100 kΩ pull-down resistor, which can measure the contact force from 0 N to 4 N using a 3 V, 12-bit ADC bus.

3.1 Current–Force Conversion. The proposed exoskeleton glove does not feature a force sensor on all fingers; therefore, it is challenging to control force. We controlled the motor force output indirectly by controlling the electrical current consumption of the motor. The conversion of current to force is discussed in this subsection.

To perform the linear actuator current–force calibration, we placed a load cell to measure the force exerted by the last linkage of the middle finger mechanism (see Fig. 5(*a*)). Force is measured by the load cell in the perpendicular direction to the ground. The measured force on the load cell can be modeled using Eq. (18)

$$F_{lc} = F_m \cdot R(p) \cdot \cos(\theta) \tag{18}$$

where F_{lc} denotes the force measured by load cell, F_m denotes the force output by linear actuator, $R(p)$ denotes the actuator to fingertip linkage force transformation ratio as a function of actuator position, and θ denotes the angle between load cell and fingertip linkage.

The actuator can be calibrated by mapping the current consumption I to the actuator output force F_m. I can be measured directly by the motor controller. F_m can be calculated using Eq. (18), where actuator to fingertip force transformation ratio is a function of the actuator position shown in Fig. 5(*b*). The encoder measures the actuator position p, and θ is measured by hand. We collected 135 force–current data pairs at three angles to perform the fitting and use 90 pairs to validate the result. The fitting result is shown in Fig. 5(*c*). The calibration achieved an average force difference of 21.15% mean absolute percentage error (MAPE) at the fingertip linkage. The actuator can output around 13.5 N and 1.6–2.76 N to fingertip linkage based on different linkage angles.

To perform rotatory actuator current–force calibration, we placed a load cell to measure the force exerted by the last linkage of the thumb and thumb thenar mechanism (see Fig. 6(*a*)). Force is measured by the load cell in the perpendicular direction to the ground.

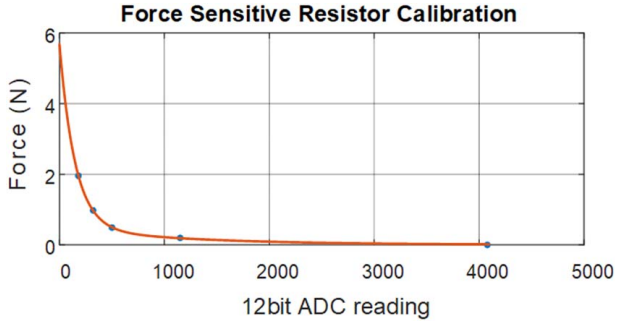

Fig. 7 Force calibration for FSR-402 force-sensitive resistor

The measured force on the load cell can be modeled using Eq. (19)

$$\tau_t = F_{lc} \cdot \cos(\theta) \cdot r \tag{19}$$

where F_{lc} is the force measured by load cell, τ_t represents the torque output by rotatory actuator, θ represents the angle between load cell plane and rotatory actuator rotational axis, and r represents rotatory actuator leverage.

The actuator can be calibrated by mapping the current consumption I to the actuator output force τ_t. I can be measured directly by the motor controller. τ_t can be calculated using Eq. (19), where θ and r are measured by hand. We collected 64 pairs of force–current data at two angles to perform the fitting and used 48 pairs to validate the result. The fitting result is shown in Fig. 6(*b*). The calibration achieved an average torque difference of 10.25% MAPE and the actuator can output around 37 N · cm torque.

3.2 Force-Sensitive Resistor Calibration. The force-sensitive resistors are placed on the index finger pad and on the inner side of the thumb finger (measure the force output from the thumb). FSRs are used to assist in force sensing, contact detection, and slip detection. The maximum force output of the index fingertip linkage is 2.76 N, and the maximum force output of the thumb rotatory actuator is 4.07 N. Therefore, 25 g, 50 g, 100 g, and 200 g weights are used to calibrate FSRs. Each weight is placed on the FSR five times. The average ADC reading has been recorded for each weight. The calibration curve is shown in Fig. 7.

3.2.1 Controller Software Architecture. This subsection provides an overview of the low-level and high-level control structures. The low-level control is executed on the Teensy 4.1 microcontroller, while the high-level control runs on a computer. The comprehensive software architecture is illustrated in Fig. 8.

The primary functions of the low-level controller encompass actuator control, sensor signal filtering and processing, and communication. Actuator control is governed by two PID controllers, selected based on whether the fingertip needs to generate force on the target object for the specific grasp type. The PID current controller is utilized to generate force on the fingertips, while the PID position controller manages the closing motion without generating any contact force on the target object. For instance, during a cylinder grasp, all six actuators employ force control, utilizing the PID current controller for actuation. In contrast, for a tip grasp, where the middle, ring, and pinky fingers maintain fixed positions, PID position control is applied to these three fingers' actuators.

Sensor signal filtering involves a bandpass filter that collects motor current consumption data at a rate of 100 Hz and FSR readings at 10 Hz. The motor current consumption data serve as feedback for the PID current controller at 100 Hz and are then converted into actuator force readings at a rate of 10 Hz. Simultaneously, the FSR readings are converted into fingertip force measurements, also at 10 Hz. These converted values, including actuator force, fingertip force, and actuator position, are transmitted to the high-level controller at a rate of 10 Hz.

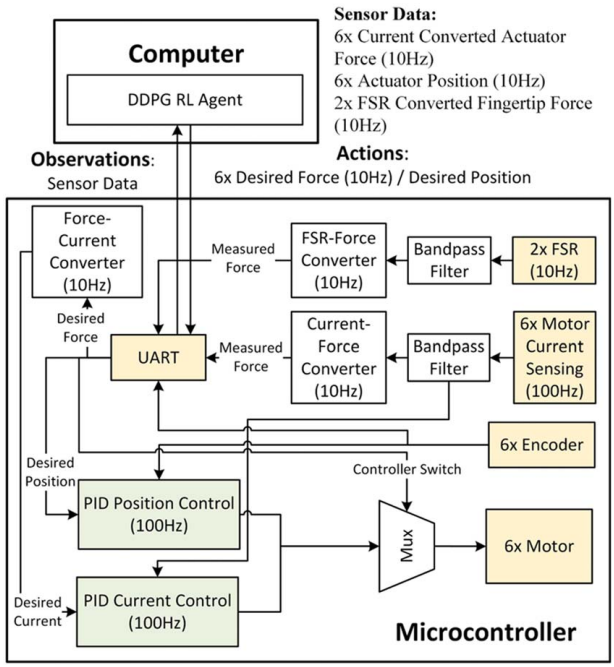

Fig. 8 Exoskeleton glove system software architecture. Hardware and peripherals are marked in yellow. Controllers are marked in green. The microcontroller sends sensor data to the computer at a rate of 10 Hz and receives force or position command from the computer at the same speed.

The high-level controller operates a DDPG deep reinforcement learning agent, as elaborated in Sec. 4. This agent takes observations from the low-level controller and sends actions back to the low-level controller to control the exoskeleton glove's operations at a rate of 10 Hz.

4 Reinforcement Learning Control Policy

The current research on exoskeleton gloves primarily focuses on mechanism design, with limited attention given to the development of force control algorithms [24]. Many existing control policies are designed to demonstrate the feasibility of the mechanisms but lack the capability to adjust forces during grasping and adapt to different grasp types [15]. Recognizing these shortcomings, we propose an adaptive force control method based on reinforcement learning. This approach aims to enable real-time force adjustment during grasping based on different target objects. By employing this method, we seek to enhance the exoskeleton glove's ability to plan forces according to the specific requirements of different grasping tasks.

4.1 Simulation Environment. Simulation plays a crucial role in verifying the functionality of both the hardware and control system, as well as accelerating the design process for complex control algorithms. The proposed exoskeleton glove simulation was conducted using MATLAB SIMSCAPE R2022b. Comparing to other simulation software such as MUJOCO, V-REP, and GAZEBO, SIMSCAPE offers a more stable solution for simulating coupling mechanisms and parallel robots. It also provides an easy way to import complex 3D models, establish connections between the simulation environment and agent, and implement real-time control based on trained agents.

The simulation can be divided into three main parts: (1) selection of simulation parameters, (2) control of simulated actuators, and (3) analysis of simulation results.

4.1.1 Simulation Parameter Selection. To ensure the proper functioning of a simulation, three sets of parameters are essential.

The first set of parameters in our simulation involved modeling the physical characteristics of the simulated objects, which included their mass, center of mass, and moment of inertia. To ensure accuracy and consistency, we automatically generated these parameters for the exoskeleton glove by assigning appropriate material properties in SOLIDWORKS.

For the target objects, their shape and weight were subjected to variations within a range of ±15% of their original size and weight (Table 3). This variation was introduced to create a diverse set of objects with slightly different properties, simulating real-world scenarios where objects may vary in size and weight.

The second set of parameters set the internal mechanisms of the simulated actuators. There are seven actuators on the physical exoskeleton glove: one for each digit, one for the thumb CMC joint, and one for the wrist (mimic the human wrist motion in the simulation environment only). To replicate the abduction and adduction motions of human fingers, passive revolute joints were used between the finger mechanism and the exoskeleton glove base. In the simulation, these passive joints are set with high stiffness and constraints to reduce computational costs. This eliminates the need for calculating contact forces between different fingers. Therefore, each finger comprises two actuators: one prismatic joint for controlling linear motion and one revolute joint for simulating abduction and adduction motions. Additionally, there is one more actuator for the rotary motion of the thumb thenar. To represent a motor in the physical hardware, a spring-damper system is utilized as the internal mechanism for these simulated actuators. The prismatic joint's spring-damper system has a spring stiffness of 0 N/m, a damping ratio of 1300 N/(m/s). The thumb rotatory joint has a spring stiffness of 0 (N · m)/deg and a damping ratio of 0.001 N · m/(deg/s). These values were determined through trial and error to match the hardware behavior.

The third set of parameters modeled the contact forces, specifically in the contact between the finger and the target object, with the aim of achieving accurate simulation result. These contact force parameters include stiffness, damping ratio, static friction, and dynamic friction. These values were obtained from previous research [25].

4.1.2 Simulated Actuator Control. Specifically, for the revolute joints responsible for simulating the abduction and adduction motions, they are controlled by a built-in position controller, which allows them to maintain specific positions depending on the grasp type. On the other hand, the motor and leadscrew combination on each digit is simulated as one linear actuator, and the linear actuators are controlled using a built-in force controller, ensuring precise force control. Lastly, the thumb rotary actuator is controlled using a built-in torque controller, enabling accurate torque control. These control strategies effectively govern the behavior of the simulated actuators in the system.

4.2 Control Policy. DRL techniques are particularly well-suited for learning in high-dimensional continuous observation and action spaces. Unlike traditional reinforcement learning methods, DRL leverages deep neural networks to map states to actions, enabling it to handle complex and continuous environments without the need for a predefined or learned dynamics model [26].

Among the model-free DRL methods, two prominent approaches are proximal policy optimization (PPO) [27] and DDPG [28]. PPO is an on-policy method that directly optimizes the policy through policy gradient techniques. It uses a trust region framework to ensure stability during policy updates and incorporates a clipped surrogate objective to avoid overly large policy changes. However, since the PPO generates the actions based on the mean and standard deviation, it is hard to enforce physical constraints on the action space. This may require additional training steps to ensure that the actions remain within the desired range. Moreover, the environment is expensive to sample from, which increase the training time.

On the other hand, DDPG belongs to the off-policy category, where it learns a deterministic policy and utilizes an off-policy experience replay buffer to improve sample efficiency and stability. It employs an actor-critic architecture, where the actor network learns the policy, and the critic network learns the value function. This approach has been shown to the success in continuous control tasks.

Our approach utilizes the DDPG algorithm to implement the force controller for the proposed exoskeleton glove. The DDPG agent is composed of an actor and a critic neural network to represent the policy and value functions, respectively. The actor network takes actions based on its current policy and observations from the environment, and the critic evaluates these actions based on the observations and a reward. The simulation environment is composed of a simulated exoskeleton model as mentioned before.

4.2.1 Reward Function. Designing a suitable reward function is one of the key challenges in training the DRL agent. The reward function must fulfill two simultaneous requirements. First, the exoskeleton should exert minimal effort while successfully lifting objects of various sizes and shapes. Second, the exoskeleton's motion should closely adhere to the position limitations and natural movements of the human hand. During a cylinder grasp, it is important that all fingers extend until they make contact with the target object. Additionally, it is desirable for the extension of each finger to remain relatively consistent, without significant variation.

In light of these considerations, the reward function can be defined as follows:

$$r = c_1 R_s + c_2 R_h + c_3 R_f + c_4 R_{fd} + c_5 R_c \tag{20}$$

where

$$R_s = -|h_e - h_o| \tag{21}$$

$$R_h = \begin{cases} 3, & \text{if lift height} \geq 3\,\text{mm} \\ 0, & \text{otherwise} \end{cases} \tag{22}$$

$$R_f = -\sum_n f_n \tag{23}$$

$$R_{fd} = -\sum_n |f_{d_n}| \tag{24}$$

$$R_c = \begin{cases} T_s/T_f, & \text{if contact is detected} \\ 0, & \text{otherwise} \end{cases} \tag{25}$$

h_e denotes the height of the exoskeleton glove, h_o denotes the height of the object, d_o signifies the lift distance of the target object, f_n denotes to the force generated by each finger actuator (where $n \in \{thumb, index, middle, ring, little\}$), f_d represents the first-order derivative of the generated force, T_s denotes the sample time, and

Table 3 Objects used in grasp experiments

Name	Weight (g)	Grasp type
Glue Jar	200	Cylinder
Water Bottle	600	Cylinder
Orange	385	Sphere
Tennis Ball	69	Sphere
Pen	13	Tripod
Flat Box	102	Lateral
Paper Bowl	8	Lateral
Small Box	11	Tip
Heavy Box	211	Tip

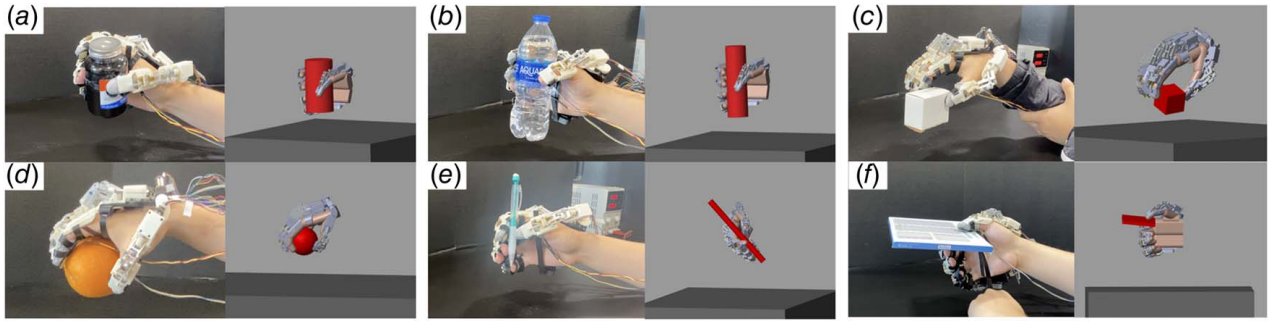

Fig. 9 Successful grasps of various objects in both simulation environment and real-world: (*a*) jar, (*b*) bottle, (*c*) small box, (*d*) orange, (*e*) pen, and (*f*) flat box

T_f denotes the final simulation time of the environment.

The term R_s is utilized to penalize the agent in the condition of slip occurrence, while R_h encourages the agent to successfully lift the target object. To promote minimal effort in lifting, the term R_f penalizes the force generated by each actuator. In order to enhance consistency in the generated force, R_{fd} penalizes significant variations in force. Additionally, R_t is incorporated to motivate the agent to avoid premature termination.

4.2.2 Observation and Action Spaces. The observation space determines the information available to the robot agent for policy determination. At each time-step, the robot agent receives the following observations: $\{p_n, q_n, v_n, c_i\}$, where p denotes the position of the linear actuator, q denotes the force generated from linear actuator, v denotes the speed of the linear actuator, $n \in \{thumb, index, middle, ring, little\}$, c denotes the contact information on the thumb and index fingertips, and $i \in \{thumb, index\}$. These specific observations were selected based on the measurable characteristics of the exoskeleton glove in real-world scenarios. To ensure consistency and compatibility, all observations are normalized within a range of 0–1 before being inputted to the DDPG agent for training and decision-making.

The action space in this exoskeleton glove is relatively straightforward, as it corresponds to the number of actuators present. The deep neural network is trained to generate motor control commands, and any error correction or adjustment is learned by the policy network itself. To ensure that the output of the actor neural network falls within the desired range, a sigmoid function is applied to the last layer. This sigmoid function restricts the output to the range of 0–1. By multiplying this bounded action with the appropriate force factor, the output can be easily converted to the desired force control range, which is from 0 N to 13 N for fingers and from 0 N · m to 0.35 N · m for the thumb in this case.

It is important to note that in the real-world scenario, the lifting motion is typically controlled by human. However, in the simulation environment, a simplified approach is taken. When the thumb and index finger make contact with the object, a constant force of 25 N is applied to simulate the lifting action. This approximation is used to mimic the lifting process observed in the real-world, but it is important to note that this lifting motion is not generated by the DDPG agent itself. Instead, it is triggered by the simulation environment to provide a realistic interaction between the exoskeleton glove and the object being lifted.

4.2.3 Episode Termination Criteria in Deep Reinforcement Learning Training. During the training process, we employ three stopping criteria to terminate each episode's simulation. First, if the object has slipped by more than 3 cm, indicating a loss of grip. Second, if the exoskeleton reaches an uncomfortable position, such as when the distance between each finger actuator exceeds 4 mm. In such cases, the fingers are separated, resulting in an unnatural gesture. Lastly, if the simulation time surpasses a predefined limit, the episode is terminated.

5 Experiments

The training process was conducted on a machine equipped with an AMD 3700X CPU @ 3.60 GHz, 32 GB of memory, and a Nvidia GeForce RTX 2080TI GPU. The entire training process took approximately 60 h to complete. To showcase the exoskeleton glove's capabilities, a series of grasp experiments were conducted both in a simulation environment and using the physical exoskeleton glove. Nine target objects (listed in Table 3) were selected for performing five different types of grasp: cylinder grasp, sphere grasp, tripod grasp, lateral grasp, and tip grasp. The experiments will be discussed from three perspectives: matching the simulation environment with the physical experiment, simulation results, and physical experiments with the exoskeleton glove hardware.

5.1 Bridging the Gap Between Simulation and Physical Environment. In order to bridge the gap between the simulation and physical environments, efforts were made to closely match the observations and actions of the reinforcement learning agent to minimize the disparities between the two environments. The agent's observations include actuator position, speed, force, and fingertip force. The position and speed of the linear actuators can be directly measured. The calibrated FSR force mentioned in Sec. 3 can be used as the fingertip force in the simulation environment. Furthermore, the motor force calibrated using motor current consumption, as described in Sec. 3, can be employed as the actuator force in the simulation environment. The actuator force serves as the control output from the simulation environment and is converted to motor current using the calibration results outlined in Sec. 3.

5.2 Simulation Results. In order to validate the effectiveness of the trained reinforcement learning agent in performing grasping tasks, a series of grasp experiments were conducted in a simulation environment. These experiments aimed to ensure that the agent could achieve successful grasps with appropriate finger positions and actuator forces. Figure 9 demonstrates the reinforcement learning agent successfully grasping the target objects in the simulation environment.

5.3 Experiments in Physical Environment. The reinforcement learning agent was successfully implemented and tested on the physical exoskeleton glove using a healthy subject as a participant in the experiments. We chose to use a human subject instead of a wooden hand for the experiment due to the wooden hand's lack of abduction and adduction motions. Moreover, the exoskeleton glove's rigid linkage construction prevents the subjects from exerting force on the exoskeleton, making the use of a healthy subject appropriate and justifiable. It is important to note that the agent's assistance is primarily focused on the grasping motion. During the releasing motion, the exoskeleton glove will move at a

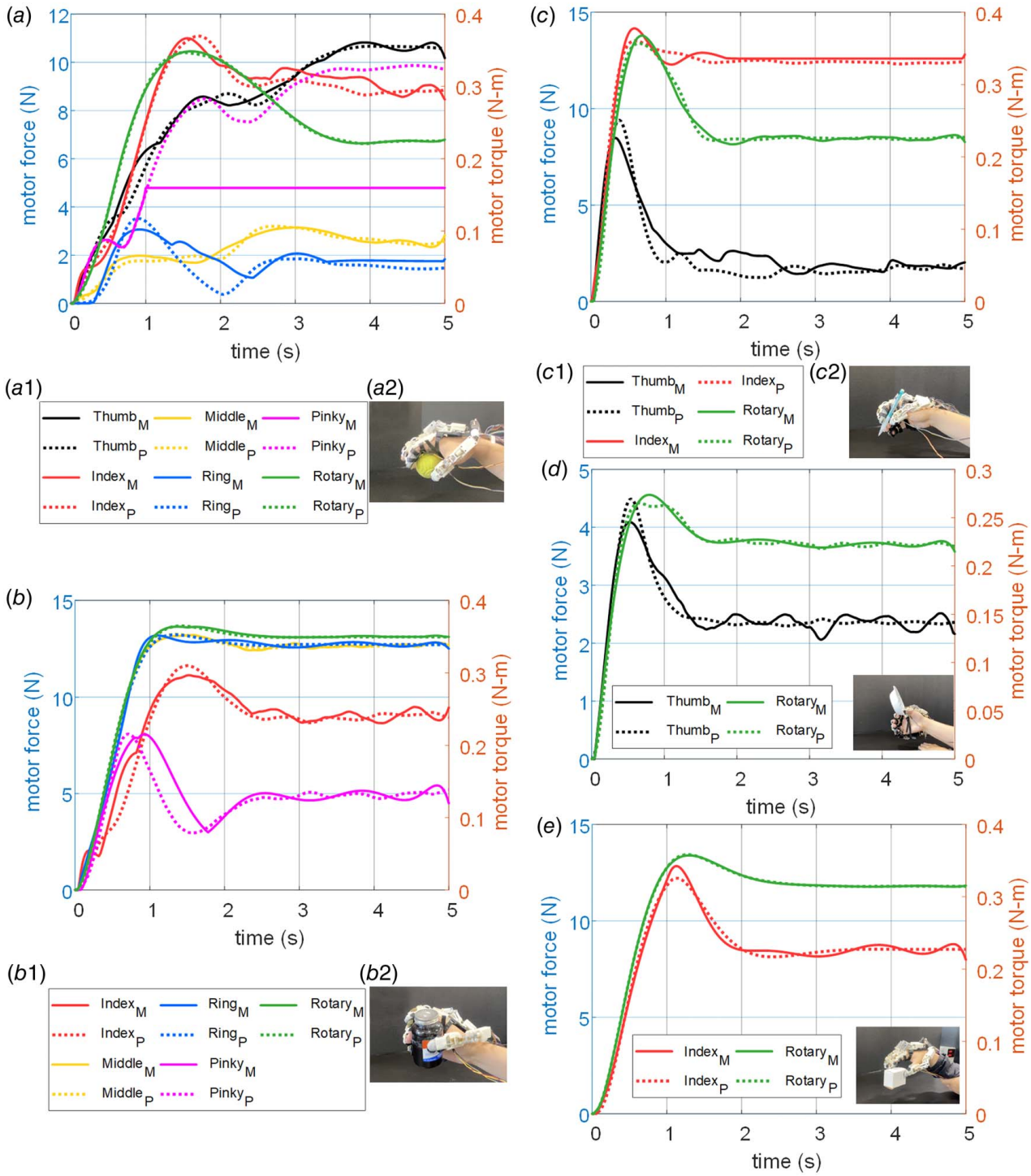

Fig. 10 The comparison between planned actuator force and measured actuator force when performing five different grasp types. The subscript P denotes the planned force/torque (shown as dashed line) on the actuators based on the trained DDPG agent. The subscript M denotes the calculated force/torque (shown as solid line) on the actuators based on the calibration results. The left *y*-axis indicates the force generated by the linear actuators, while the right *y*-axis represents the torque generated by the rotary actuator on the thumb. Five different grasp types were performed: (*a*) sphere grasp (pinky finger hit control limit), (*b*) cylinder grasp, (*c*) tripod grasp, (*d*) lateral grasp, and (*e*) tip grasp.

predefined constant speed to ensure patient safety within a short timeframe.

In Fig. 10, we present the comparison between the measured force and the planned force generated by the reinforcement learning agent during the grasp experiments. It is worth mentioning that we only display the curve corresponding to the digits or CMC joint that is sensed with normal force for clarity and simplicity. During the experiment, the other actuators move to their end position at full speed. In the sphere grasp scenario, the actual object being smaller than the trained object results in the pinky finger reaching its position limit before generating the planned full force.

The results clearly demonstrate that the measured force closely tracks the planned force generated by the reinforcement learning agent, except when the control limit is reached. This close alignment between the planned and measured forces highlights the agent's exceptional ability to accurately control the exoskeleton's

force output. The agent's precise and reliable grasp control during the experiments further emphasizes its potential for assisting individuals with hand disabilities in performing various daily living tasks.

6 Conclusion And Future Work

In this paper, we presented a novel exoskeleton glove designed to assist individuals with hand disabilities in their activities of daily living. The glove features a low-profile, compact, and portable design, making it suitable for everyday use. We introduced the mechanism and mechanical design of the glove, focusing on achieving a better alignment with the natural motions of the human hand through kinematic analysis and synthesis. To demonstrate the feasibility of our design, we developed a proof-of-concept prototype and implemented the necessary electronic components and low-level control policy. Additionally, we used the deep reinforcement learning to provide real-time self-adjustable force control based on different objects. To train our control policies, we created a simulation environment that allowed us to simulate various scenarios and train the agents effectively. Through several experiments, we showcased the capabilities of our exoskeleton glove in grasping daily objects with dexterity.

However, we also identified some issues with the existing prototype. The limited DOFs of the glove restrict its ability to handle more complex grasping tasks. Additionally, there is a noticeable gap between the simulation environment and the real-world, making it challenging to achieve a close match between them. The predefined mass-spring-damper model in MATLAB lacks the properties of actual linear actuators, leading to discrepancies between the simulated and real-world behaviors. Additionally, the reinforcement learning agent may not consistently provide a constant force during stable grasps. This variability is attributed to the noise intentionally added to the agent's actions during training to promote exploration. While the noise is essential for learning and adapting to different scenarios, it can lead to less consistent force control during practical applications, especially for tasks requiring precise and steady force application. Furthermore, finding compact and elaborate force sensors for precise control remains a challenge. Other electronics could be integrated on the forearm, thereby increasing the whole system's portability. Besides, the incorporation with the hand for bimanual tasks can be highly beneficial for enhancing patients' daily activities. To move forward, addressing these challenges will be the main focus of our future work and we believe that this exoskeleton glove will provide valuable assistance to individuals with hand disabilities.

Acknowledgment

Research reported in this publication was supported by the Eunice Kennedy Shriver National Institute of Child Health & Human Development of the National Institutes of Health under Award Number R21HD095027. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. This material is based upon work supported by (while serving at) the National Science Foundation.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The authors attest that all data for this study are included in the paper.

References

[1] Park, H. R., Lee, G. S., Kim, I. S., and Chang, J. -C., 2017, "Brachial Plexus Injury in Adults," The Nerve, **3**(1), pp. 1–11.

[2] Giuffre, J. L., Kakar, S., Bishop, A. T., Spinner, R. J., and Shin, A. Y., 2010, "Current Concepts of the Treatment of Adult Brachial Plexus Injuries," J. Hand Surg., **35**(4), pp. 678–688.

[3] Ferguson, P. W., Dimapasoc, B., Shen, Y., and Rosen, J., 2018, "Design of a Hand Exoskeleton for Use With Upper Limb Exoskeletons," 2018 International Symposium on Wearable Robotics and Rehabilitation, Pisa, Italy, Oct. 16–20, Springer, pp. 276–280.

[4] Iqbal, J., Tsagarakis, N., Fiorilla, A. E., and Caldwell, D., 2009, "Design Requirements of a Hand Exoskeleton Robotic Device," 14th IASTED International Conference on Robotics and Applications (RA), Cambridge, MA, Nov. 2–4, Vol. 64, pp. 44–51.

[5] Polygerinos, P., Wang, Z., Galloway, K. C., Wood, R. J., and Walsh, C. J., 2015, "Soft Robotic Glove for Combined Assistance and At-Home Rehabilitation," Rob. Auton. Syst., **73**, pp. 135–143.

[6] Kudo, S., Oshima, K., Arizono, M., Hayashi, Y., and Moromugi, S., 2014, "Electric-Powered Glove for CCI Patients to Extend Their Upper-Extremity Function," 2014 IEEE/SICE International Symposium on System Integration, Tokyo, Japan, Dec. 13–15, Institute of Electrical and Electronics Engineers Inc., pp. 638–643.

[7] Yap, H. K., Ang, B. W., Lim, J. H., Goh, J. C., and Yeow, C. H., 2016, "A Fabric-Regulated Soft Robotic Glove With User Intent Detection Using EMG and RFID for Hand Assistive Application," 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, May 16–21, Institute of Electrical and Electronics Engineers Inc., pp. 3537–3542.

[8] Borboni, A., Mor, M., and Faglia, R., 2016, "Gloreha-Hand Robotic Rehabilitation: Design, Mechanical Model, and Experiments," ASME J. Dyn. Syst., Meas. Contr., **138**(11), p. 111003.

[9] Ma, Z., Ben-Tzvi, P., and Danoff, J., 2015, "Sensing and Force-Feedback Exoskeleton Robotic (SAFER) Glove Mechanism for Hand Rehabilitation," International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Boston, MA, Aug. 2–5, pp. 1–8.

[10] Hofmann, U. A., Bützer, T., Lambercy, O., and Gassert, R., 2018, "Design and Evaluation of a Bowden-Cable-Based Remote Actuation System for Wearable Robotics," IEEE Rob. Autom. Lett., **3**(3), pp. 2101–2108.

[11] Leonardis, D., Barsotti, M., Loconsole, C., Solazzi, M., Troncossi, M., Mazzotti, C., Castelli, V. P., Procopio, C., Lamola, G., Chisari, C., Bergamasco, M., and Frisoli, A., 2015, "An EMG-controlled Robotic Hand Exoskeleton for Bilateral Rehabilitation," IEEE Trans. Haptics, **8**(2), pp. 140–151.

[12] Ho, N. S., Tong, K. Y., Hu, X. L., Fung, K. L., Wei, X. J., Rong, W., and Susanto, E. A., 2011, "An EMG-Driven Exoskeleton Hand Robotic Training Device on Chronic Stroke Subjects: Task Training System for Stroke Rehabilitation," 2011 IEEE International Conference on Rehabilitation Robotics, Zurich, Switzerland, June 29–July 1.

[13] Xu, W., Guo, Y., Bravo, C., and Ben-Tzvi, P., 2022, "Design, Control, and Experimental Evaluation of a Novel Robotic Glove System for Patients With Brachial Plexus Injuries," IEEE Trans. Rob., **39**(2), pp. 1637–1652.

[14] Xu, W., Guo, Y., Bravo, C., and Ben-Tzvi, P., 2022, "Development and Experimental Evaluation of a Novel Portable Haptic Robotic Exoskeleton Glove System for Patients With Brachial Plexus Injuries," 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, Oct. 23–27, IEEE, pp. 11115–11120.

[15] Tran, P., Jeong, S., Herrin, K. R., and Desai, J. P., 2021, "Hand Exoskeleton Systems, Clinical Rehabilitation Practices, and Future Prospects," IEEE Trans. Med. Rob. Bionics, **3**(3), pp. 606–622.

[16] Xu, W., Liu, Y., and Ben-Tzvi, P., 2022, "Development of a Novel Low-Profile Robotic Exoskeleton Glove for Patients With Brachial Plexus Injuries," 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, Oct. 23–27, IEEE, pp. 11121–11126.

[17] Bekey, G. A., Tomovic, R., and Zeljkovic, I., 1990, "Control Architecture for the Belgrade/USC Hand," *Dextrous Robot Hands*, S. T. Venkataraman, and T. Iberall, eds., Springer, New York, NY, pp. 136–149.

[18] Ma, Z., Ben-Tzvi, P., and Danoff, J., 2016, "Hand Rehabilitation Learning System With an Exoskeleton Robotic Glove," IEEE Trans. Neural. Syst. Rehabil. Eng., **24**(12), pp. 1323–1332.

[19] Refour, E., Sebastian, B., and Ben-Tzvi, P., 2018, "Two-Digit Robotic Exoskeleton Glove Mechanism: Design and Integration," ASME J. Mech. Rob., **10**(2), p. 025002.

[20] Liu, Y., and Ben-Tzvi, P., 2020, "Design, Analysis, and Integration of a New Two-Degree-of-Freedom Articulated Multi-link Robotic Tail Mechanism," ASME J. Mech. Rob., **12**(2), p. 021101.

[21] Xu, W., Liu, Y., and Ben-Tzvi, P., 2022, "Design, Analysis, and Prototyping of a Novel Single Degree-of-Freedom Index Finger Exoskeleton Mechanism," International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Vol. 86281, American Society of Mechanical Engineers, p. V007T07A047.

[22] Della Santina, C., Bianchi, M., Averta, G., Ciotti, S., Arapi, V., Fani, S., Battaglia, E., Catalano, M. G., Santello, M., and Bicchi, A., 2017, "Postural Hand Synergies During Environmental Constraint Exploitation," Front. Neurorobot., **11**(August), pp. 1–14.

[23] Lin, J., Wu, Y., and Huang, T. S., 2000, "Modeling the Constraints of Human Hand Motion," Proceedings Workshop on Human Motion, Austin, TX, Dec. 7–8, pp. 121–126.

[24] Ferguson, P. W., Shen, Y., and Rosen, J., 2020, "Hand Exoskeleton Systems-Overview," *Wearable Robotics*, J. Rosen, and P. W. Ferguson, eds., Academic Press, Cambridge, MA, pp. 149–175.
[25] Park, J., Pažin, N., Friedman, J., Zatsiorsky, V. M., and Latash, M. L., 2014, "Mechanical Properties of the Human Hand Digits: Age-Related Differences," Clin. Biomechan., **29**(2), pp. 129–137.
[26] Di Febbo, D., Ambrosini, E., Pirotta, M., Rojas, E., Restelli, M., Pedrocchi, A. L., and Ferrante, S., 2018, "Does Reinforcement Learning Outperform PID in the Control of FES-Induced Elbow Flex-Extension," 2018 IEEE International Symposium on Medical Measurements and Applications (MeMeA), Rome, Italy, June 11–13, IEEE, pp. 1–6.
[27] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O., 2017, "Proximal Policy Optimization Algorithms," arXiv preprint arXiv:1707.06347.
[28] Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D., 2015, "Continuous Control With Deep Reinforcement Learning," arXiv preprint arXiv:1509.02971.