Fuzzy CMAC-Based Adaptive Scale Force Control of Body Weight Support Exoskeletons

HAN WU, HONGLEI AN, QING WEI, AND HONGXU MA
Robot Research Center, College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China
Corresponding author: Honglei An (wuhan10@nudt.edu.cn)
This work was supported by the National Key Research and Development Program of China under Grant 2018YFC2001304.

ABSTRACT The body weight support method of exoskeletons is useful for the rehabilitation of patients with lower limb dyskinesia and performance augmentation of elders. Rather than enhancing the human joint strength directly, the body weight support exoskeleton provides assistance by supporting part of the human body weight and thus releasing the burden experienced by human legs. This paper proposes a two-level hierarchical control method aiming to achieve task-free and strong system robustness. A scale force control method, which can be simply described as making the supporting force of the robot several times the magnitude of a human leg, is developed as the high-level controller. The low-level controller is designed as a fuzzy cerebellar model articulation controller (CMAC)-based adaptive control to enhance the system robustness. The proposed method is applied to a squat movement and a single leg stance movement and is simulated in a combined simulation environment. The results show that the exoskeleton can always provide improved support according to the foot reaction force that is controlled by the user and that the system handles model errors and external disturbances well.

INDEX TERMS Body weight support, exoskeleton, fuzzy CMAC, scale force control.

I. INTRODUCTION
An exoskeleton is a wearable device that can increase the performance of able-bodied wearers [1], [2]. There are many control strategies for achieving assistance, such as sensitivity amplification control (SAC) [3], [4], electromyography (EMG)-based control [5]–[7], and trajectory tracking control [8]. However, few strategies have a good clinical use effect. The body weight support (BWS) control method provides a useful technique and is mainly used for reducing lower limb burden of patients or elders [9]–[11]. Although there are already some devices that achieve the function of BWS, Locomat [12], [13] for example, they are very large and bulky. BWS control methods for portable exoskeletons are still under investigation.

The earliest BWS device resembling an exoskeleton can be traced back to 1890 [14]. The device is passive and supports the wearer’s weight with a pair of long bow springs. Research on passive BWS devices has continued in recent years. Lee designed a passive BWS lower limb exoskeleton with a bionic knee joint for walking gait [15], [16]. Krut developed a quasi-passive exoskeleton, called MoonWalker, to lighten the forces in legs due to bodyweight [17]. The passive BWS exoskeleton can always provide an anti-gravity support force for the wearer, but the assistance characteristic is inherent, and the passive devices cannot follow the user’s movement intentions.

Powered BWS exoskeletons can provide personalized assistance. Compared with powered exoskeletons that provide assistance on joints directly, there are few studies of portable powered BWS exoskeletons. Lv achieves partial body weight assistance when walking by using a control strategy based on underactuated potential energy shaping [18]. However, the potential energy shaping method relies heavily on the motion modes and phase divisions. Honda developed a walking assist device that can be sat on [19]. The structure of the device is designed based on a biomedical engineering analysis so that it can always maintain the assist force vector in the direction from the center of pressure of the floor reaction force to the center of gravity of the user’s body. The desired assist force of Honda’s walking assist device is constant and can be user-defined. Then, the assist force is distributed to two legs according to the foot reaction forces on each leg. A similar device can be seen in [20]. However,
constant BWS assistance cannot be user-friendly; for example, the user needs substantial assistance when going up stairs and minimal assistance while walking on a flat floor. A more automated controller is needed that can make the desired assist force be related to the state of human motion rather than a constant value. Mir-Nasiri developed a conceptual design of exoskeleton that supports the user’s body weight by locking the knee joint in some stance phases [21]. All the above powered BWS exoskeletons have little consideration of the wearer, that is, the users cannot totally control the robot. In this paper, we hope that the exoskeleton has an effect that works like additional human leg muscles and bearing loads (the user body weight) that are several times larger than the user’s legs bear. By using this method, the wearer only uses a scaled down effort to achieve the same movement and can completely handle the device regardless of its motion modes.

In addition to the control target selection of the BWS exoskeleton, another challenge is the robustness of the control system, which is important for the safety of the human-robot system. The most commonly used robust control methods include slide mode control [22], [23], fuzzy-based adaptive control [24], [25], and neural network-based adaptive control [26], [27]. Among them, neural network-based adaptive control has the advantage of approximating the unknown nonlinear characteristics of the system dynamics. Generally, the BWS exoskeleton is a kind of biped exoskeleton. To simplify the modeling and control of the system, decoupling is needed for the stance leg and the swing leg. Thus, for the stance leg, the motion of the swing leg can be regarded as external disturbances, which also requests robustness of the BWS control method of the stance leg.

Aiming to resolve the above two problems of the BWS exoskeleton, namely, task-free control target selection and strong robustness controller design, we provide a hierarchical control method [28] for loosely coupled exoskeletons by using sufficient interaction force information. The control method is a kind of assist-as-needed method rather than a constant assistance method. The assistance varies automatically with human movement and the assistance level can be adjusted freely by the user. Additionally, the control method has strong system robustness which increases the system security.

The general hierarchical control framework includes three layers: a high-level layer for motion mode recognition, a mid-level layer for desired trajectory generation, and a low-level layer for actuator control. In this work, only two layers are sufficient. The high-level control target is selected to control the BWS force (can be measured by the foot reaction force), and we bears (can be measured by the foot reaction force), and we called this method scale force control (SFC). SFC ensures that the robot provides variable assistance along with changes in human motion. For the low-level layer, a fuzzy cerebellar model articulation controller (CMAC) [29]–[31]-based adaptive controller is designed. Simulations indicate that fuzzy CMAC-based adaptive scale force control is capable of providing any interested body weight assistance only using interaction forces, and the system has a strong tolerance for model uncertainties and outside disturbances.

The rest of this paper is organized as follows. Section 2 introduces the concept of the loosely coupled BWS exoskeleton and builds the model of it. Section 3 describes the proposed control method in detail. The simulation and results are presented in Section 4. Finally, Section 5 summarizes the conclusions, discusses related problems and gives an outlook of the future work.

II. SYSTEM DESCRIPTION AND DYNAMIC MODELING

Many exoskeleton structures are designed to be anthropomorphic or semi-anthropomorphic, such as HAL [32], ReWalk [33], and MINDWALKER [34]. Their mechanical arms are parallel with human limbs so that the robot can provide specific assistance for each human joint. However, that is quite a challenge because of the complexity of human joint structures and individual differences. Another kind of exoskeleton provides assistance from the overall effect through a structure that is loosely coupled with that of a human. For example, MoonWalker [17] only connects at the foot and waist and supplies assistance for the entire lower limb. The benefits of the loosely coupled exoskeleton include a simple structure, wearer convenient, less human-machine interference and so on. In this work, we used a loosely coupled lower limb exoskeleton to achieve body weight support function.

A. SYSTEM DESCRIPTION

The simulation model (Fig. 1) is designed in the combined simulation environment of MATLAB and OpenSim [35]. The later software provides various anatomical models of the human body. Scripting with MATLAB allows us to interact with the classes of the OpenSim C++ API. The OpenSim-MATLAB link is established by calling functions and classes in the OpenSim API libraries. The geometry of the parts of the exoskeleton is drawn in SolidWorks. The physical parameters of the parts, e.g., mass and inertia, are defined in MATLAB. First, the human model provided by OpenSim is loaded into MATLAB. Then, we add each exoskeleton part to the human model by creating suitable joints.

The loosely coupled lower limb exoskeleton only connects to the foot and back of the wearer. At the back interaction
point, the human interacts with the exoskeleton through a spring damper. At the foot interaction locations, we assume that the interaction models are spring dampers. In the simulation model, spring-damper components and interaction force/torque sensors are added at all of the interaction points. The bending direction of the knee joint is opposite to that of the wearer to avoid human-robot interference, although it may increase the possibility of interference with the working environment. Long robot limbs are suitable for different individual statures. There are six degrees of freedom per leg in total, with four powered degrees (hip adduction/abduction, hip flexion/extension, knee flexion/extension and ankle plantar/dorsal flexion) and two passive joints (hip rotation and ankle adduction/abduction). All the powered actuators are located around the waist and equipped with encoders. The torques of the knee and ankle joints are transmitted through cables. Compared with MoonWalker and Honda’s walking assist device, the proposed exoskeleton has more actuated joints and is less bound by the structure design but has a more complex controller design. For example, for the proposed exoskeleton, the direction and size of the assist force should be controlled, while for Honda’s walking assist device, only the size of the assist force is controlled.

In this work, for simplicity, only movements in the sagittal plane are considered. The controller of the robot is based on interaction forces; thus, multidimensional force information around the interaction places is needed.

B. EXOSKELETON DYNAMICS

For body weight support exoskeletons, the stance is mainly used to provide assistance, and the swing leg needs to follow the wearer’s movements without hindering movement. Thus, we modeled the supporting leg and the swing leg separately. As Fig. 2 shows, the center of the stance foot is selected as the origin of the reference coordinate. The stance leg dynamics can be described in a general form as

$$M_{st}(q)\ddot{q} + C_{st}(q, \dot{q})\dot{q} + G_{st}(q) = T_a + J_f^T F_f + D$$

where \( q = [q_1, q_2, q_3]^T \) is the joint angular position, \( M_{st}(q) \in \mathbb{R}^{3 \times 3} \) is a symmetric positive definite inertia matrix, \( C_{st}(q, \dot{q}) \in \mathbb{R}^{3 \times 3} \) is a Coriolis and centrifugal matrix, \( G_{st} \in \mathbb{R}^{3 \times 1} \) is a gravity term, \( T_a \in \mathbb{R}^{3 \times 1} \) is the actuators torque, \( J_b \) is the Jacobian matrix, \( F_b \in \mathbb{R}^{3 \times 1} \) is the back interaction force, and \( D \) is the outside disturbance.

For the swing leg, the hip joint is defined as the coordinate origin, and the swing leg dynamics can be written as

$$M_{sw}(q)\ddot{q} + C_{sw}(q, \dot{q})\dot{q} + G_{sw}(q) = T_a + J_f^T F_f$$

where \( q = [q_4, q_5, q_6]^T \) is the joint angular position, and \( F_f \in \mathbb{R}^{3 \times 1} \) is the foot interaction force.

For loosely coupled lower limb exoskeletons, another advantage is that the physical human robot interaction can be optionally set. In this work, we use a parallel spring and damper to connect the human and the robot. The physical human robot interaction model is expressed as

$$F_b = k_b(r_{hb} - \dot{r}_{eb}) + b_b(\dot{r}_{hb} - \dot{r}_{eb})$$
$$F_f = k_f(r_{ef} - \dot{r}_{ef}) + b_f(\dot{r}_{ef} - \dot{r}_{ef})$$

in which \( k_b \) and \( k_f \) are elastic coefficients, \( b_b \) and \( b_f \) are damping coefficients, \( r_{hb} \) is the human back interaction position in the stance leg reference coordinate system, \( r_{ef} \) is the robot back interaction position in the stance leg reference coordinate system, \( r_{hf} \) is the human foot interaction position in the swing leg reference coordinate system, and \( r_{ef} \) is the robot foot interaction position in the swing leg reference coordinate system.

III. FUZZY CMAC-BASED ADAPTIVE SCALE FORCE CONTROL

The controller of the exoskeleton is designed as a two-layer structure. The high-level controller is designed using interaction forces/torques between the human and robot and load forces/torques. As an ideal control result with the high-level controller, the wearer can support the load (or their body weight) only using a scaled down effort. Thus, we defined the interaction force-based control method as scale force control. For the lower-level controller, a fuzzy CMAC-based adaptive controller is used in order to increase the system robustness.

A. SCALE FORCE CONTROL

For exoskeleton systems in which the wearer is in direct contact with the load, the load is handled under the cooperation of the human and machine. Assuming \( F_p \) is the generalized interaction force, which can be regarded as the robot assistance to the wearer, \( F_h \) is the generalized human lifting force, and \( F_f \) is the generalized load force, then the system relation can be easily given as

$$F_p + F_h = F_f$$

We expect the robot to carry the majority of the load. Assuming the robot assistance is \( K \) times the magnitude of the wearer’s effort:

$$F_p = K \cdot F_h$$
where $K > 0$ is a constant. Since it is quite difficult to measure human information directly and accurately, $F_h$ is eliminated by substituting (5) to (6) yields

$$F_p = \frac{K}{K+1} \cdot F_l$$

(7)

Thus, we obtain the scale force control law as (7) shows. The constant $K$ is called the scaling factor.

For body weight support exoskeletons, the load becomes complicated. When the system is static, the load $F_l$ is equal to the wearer’s body weight. However, when the system is dynamic, the load information is hard to calculate. Fortunately, we can obtain the wearer’s effort $F_h$ indirectly by using the foot-ground reaction force (FRF) $F_g$. Therefore, the scale force control law of the body weight support exoskeleton has the following formula

$$F_p = K \cdot F_g$$

(8)

where $F_g$ can be measured by the footplate force sensors and $F_p$ can be measured by the back force sensors. A benefit of the loosely coupled exoskeleton with scale force control is that we do not have to consider the human joint state, which is hard to accurately measure.

B. FUZZY CMAC

As shown in Fig. 3, the CMAC is constructed of five spaces: the input space $S$, association space $A$, receptive field space $T$, adjustable weight space $W$, and output space $Y$. The architecture of the fuzzy CMAC is the same as the CMAC except that the former uses the Gaussian basis function, rather than a local constant binary function, as the receptive field basis function. The progress of the network includes quantization, association, mapping to the receptive field, weighting, and output.

For a $n$-dimensional input vector $X = [x_1, x_2, \ldots, x_n]^T$, each data point in the vector should be quantized into a nondimensional section with the same range $n_i$ and minimum scale $n_i$. Usually, the minimum scale is set to $n_i = 1$. Then, each quantized input data point is represented by multilayer blocks in the association space. The association space of each input data is constructed with $n_l$ layers, and each layer is divided into $n_b$ blocks. The blocks in each layer are mismatched by one minimum scale. Assuming that the maximum block length is $l_b$, we have

$$n_l = l_b, \quad n_b = (n_x - 1)/l_b + 1$$

(9)

and there are $n_B = n_1 n_B$ blocks in total for each input. Thus, each input data is represented by $n_l$ blocks. Note that each block has a Gaussian basis function expressed as

$$\phi_{ijk}(s_i) = \exp\left[\frac{-(s_i - m_{ijk})^2}{\sigma_{ijk}^2}\right]$$

(10)

in which $i$ is the input data number, $j$ is the layer number of each input data, $k$ is the block number of each layer, and $s_i$ is the quantized input data. After that, all the blocks are recombined, and the same layer blocks from different input dimensions are tied up to constitute hypercubes in the receptive field. Thus, the input vector $X$ is converted into $n_l$ hypercubes. The $j$th layer hypercube function is defined as

$$b_{Hj}(s, m, \sigma) = \prod_{i=1}^{n} \phi_{ijk}$$

(11)

There are $n_B = n_l \times n_B$ hypercubes in total in the receptive field. Each hypercube corresponds to a weight for an output. That is, the weight space $W$ is $p$ times as large as the receptive field space if the output dimension is $p$. However, the output $y$ is only related to the selected $n_l$ hypercubes in each algorithm step:

$$y = [y_1, y_2, \ldots, y_p]^T = w^T \Gamma(s, m, \sigma)$$

(12)

where $w \in \mathbb{R}^{n_l \times p}$ is the weight matrix of the selected hypercubes, and $\Gamma(s, m, \sigma) \in \mathbb{R}^{n_l \times 1}$ is the function vector of the selected hypercubes.

C. FUZZY CMAC-BASED ADAPTIVE CONTROL

For body weight support exoskeletons, more attention is given to the stance leg control than the swing leg control. The dynamic equation (1) is expressed in the workspace as follows:

$$A(q)\dot{v}_{eb} + B(q, \dot{q})v_{eb} + G_{eb}(q) = T_a + J_b^T F_b + D$$

(13)

$$A(q) = M_a(q)J_b^{-1}(q)$$

(14)

$$B(q, \dot{q}) = C_{eb}(q, \dot{q})J_b^{-1}(q) - A(q)J_b(q, \dot{q})J_b^{-1}(q)$$

(15)
The low-level controller of the stance leg expressed in (13) can be designed by using inverse dynamics:

\[ T_u = \dot{\hat{A}}(q)\dot{q} + \dot{\hat{B}}(q, \dot{q})\ddot{r}_{eb} + \hat{G}_{st}(q) - J^T_b F_b + u_n \]

\[ \tau = \ddot{r}_{eb}^d + K^T E \]  

(16)  
(17)

where \(\hat{A}(q), \dot{\hat{B}}(q, \dot{q}),\) and \(\hat{G}_{st}(q)\) are the estimations of \(A(q), \dot{B}(q, \dot{q}),\) and \(G(q),\) respectively. \(u_n\) is the output of the neural network. \(\ddot{r}_{eb}^d\) is the desired trajectory of the interaction point of the back of the exoskeleton. \(K = [K_1, K_2]^T,\) and \(K_1\) and \(K_2\) are diagonal matrices of constants greater than zero. \(E = [r_{eb}^d - r_{eb}, \dot{r}_{eb}^d - \dot{r}_{eb}]^T\) is the error in the interaction point positions and velocities. Substituting equations (16) and (17) into (13), the stance leg dynamic equation is updated as:

\[
\Delta \ddot{r}_{eb} + K_2 \Delta \dot{r}_{eb} + K_1 \Delta r_{eb} = \hat{A}(q)^{-1} \left[ \hat{A}(q) \ddot{r}_{eb} + \ddot{\hat{B}}(q, \dot{q}) \dot{r}_{eb} + \hat{G}_{st}(q) - D - u_n \right]
\]  

(18)

where \(\Delta r_{eb} = r_{eb}^d - r_{eb}, \Delta \dot{r}_{eb} = \dot{r}_{eb}^d - \dot{r}_{eb}, \hat{A}(q), \ddot{\hat{B}}(q, \dot{q}),\) and \(\hat{G}_{st}(q)\) are modeling errors. If the right-hand side of (18) equals zero, the position error \(\Delta r_{eb}\) will converge to zero. Therefore, the next step is to design a fuzzy CMAC neural network and make the output \(u_n\) approach the sum of the model errors and external disturbances. Let

\[ \Psi = \hat{A}(q) \ddot{r}_{eb} + \ddot{\hat{B}}(q, \dot{q}) \dot{r}_{eb} + \hat{G}_{st}(q) - D - u_n \]

(19)

The stance leg state space dynamic model can be written as:

\[ \dot{\hat{E}} = UE + V\Psi \]

(20)

where \(U = [0, I^{3\times 3}; -K_1, -K_2],\) and \(V = [0; \hat{A}(q)^{-1}].\) If ideal values \(\dot{w}, m^s,\) and \(\sigma^s\) exist that can make \(\Psi \rightarrow 0,\) we have

\[ u_n^* = w^s T \Gamma(s, m^s, \sigma^s) + \varepsilon(s) \]

(21)

where \(\varepsilon(s)\) is the bounded functional reconstructational error. Let \(\hat{w}, \hat{m},\) and \(\hat{\sigma}\) be the estimated values of \(w^s, m^s,\) and \(\sigma^s;\) the output of the neural network \(u_n\) is:

\[ u_n = \hat{w}^T \Gamma(s, \hat{m}, \hat{\sigma}) \]

(22)

A compensation quantity \(u_c\) is added to the output of the neural network \(u_n\) to minimize the approximation error. Thus, the approximation error \(\tilde{u}_n\) is expressed as:

\[ \tilde{u}_n = u_n^* - u_n = \hat{w}^T \Gamma^* + \hat{w}^T \dot{\Gamma} + \varepsilon - u_c \]

(23)

where \(\hat{w} = w^s - \hat{w}\) and \(\dot{\Gamma} = \Gamma^* - \dot{\Gamma}. \) \(\Gamma\) can be expanded by Taylor series as

\[ \Gamma \equiv G^T \ddot{m} + H^T \ddot{\sigma} + O_t \]

(24)

where

\[ G = \left[ \frac{\partial b_1}{\partial m}, \frac{\partial b_2}{\partial m}, \ldots, \frac{\partial b_n}{\partial m} \right]_{m=m} \]

\[ H = \left[ \frac{\partial b_1}{\partial \sigma}, \frac{\partial b_2}{\partial \sigma}, \ldots, \frac{\partial b_n}{\partial \sigma} \right]_{\sigma=\dot{\sigma}} \]

(25)  
(26)

and \(O_t \in \mathbb{R}^{n_1 \times 1}\) is the higher-order term. Substituting (23), (24) into (20) gives:

\[
\dot{\hat{E}} = UE + V\tilde{u}_n
\]

\[ = UE + V \left[ \hat{w}^T \dot{\Gamma} + \hat{w}^T (G^T \ddot{m} + H^T \ddot{\sigma}) + \eta - u_c \right]
\]  

(27)

in which \(\eta = \hat{w}^T O_t + \hat{w}^T \dot{\Gamma} + \varepsilon\) is the error of the network and is bounded. By using the Lyapunov method, the fuzzy CMAC-based adaptive control laws are directly given as:

\[ \dot{\hat{w}} = \gamma_1 \hat{w}^T PV \]

\[ \dot{\hat{m}} = \gamma_2 \hat{w}^T V^T PE \]

\[ \dot{\hat{\sigma}} = \gamma_3 \hat{w}^T V^T PE \]

\[ u_c = \hat{\eta}, \quad \hat{\eta} = \gamma_4 \hat{w}^T PV \]

(28)  
(29)  
(30)  
(31)

where \(\gamma_1, \gamma_2, \gamma_3,\) and \(\gamma_4\) are constants greater than zero. \(P\) is a symmetric positive definite matrix that fulfills the following Lyapunov equation:

\[ PU + U^T P = -I \]

(32)

To verify the stability of the controller, the Lyapunov function is defined as:

\[ V(E, \hat{w}, \hat{m}, \hat{\sigma}, t) = \frac{1}{2} \hat{w}^T PE + \frac{1}{2}\gamma_1 tr(\hat{w}^T \hat{w}) + \frac{1}{2}\gamma_2 \hat{m}^T \hat{m} \]

\[ + \frac{1}{2}\gamma_3 \hat{\sigma}^T \hat{\sigma} + \frac{1}{2}\gamma_4 \hat{\eta}^T \hat{\eta} \]

(33)

Taking the derivative of \(V\) and substituting (22) gives:

\[ \dot{V} = \frac{1}{2} \hat{w}^T PE + \frac{1}{2} \hat{w}^T \dot{P} E + \frac{1}{\gamma_1} tr(\hat{w}^T \dot{\hat{w}}) - \frac{1}{\gamma_2} \hat{m}^T \dot{\hat{m}} \]

\[ - \frac{1}{\gamma_3} \hat{\sigma}^T \dot{\hat{\sigma}} - \frac{1}{\gamma_4} \hat{\eta}^T \dot{\hat{\eta}} \]

\[ = \frac{1}{2} E^T (U^T P + PU)E + E^T PV \Psi + \frac{1}{\gamma_1} tr(\hat{w}^T \dot{\hat{w}}) \]

\[ - \frac{1}{\gamma_2} \hat{m}^T \dot{\hat{m}} - \frac{1}{\gamma_3} \hat{\sigma}^T \dot{\hat{\sigma}} - \frac{1}{\gamma_4} \hat{\eta}^T \dot{\hat{\eta}} \]

\[ = \frac{1}{2} E^T E + E^T PV (\hat{w}^T \dot{\Gamma} + \hat{w}^T C^T \dot{m} + \hat{w}^T H^T \dot{\sigma} + \eta) \]

\[ - u_c \]

\[ = \frac{1}{\gamma_1} \left[ tr(\gamma_1 \hat{w}^T PV \hat{w}^T \dot{\Gamma}) + tr(\hat{w}^T \dot{\hat{w}}) \right] \]

\[ + \frac{1}{\gamma_2} \left[ \gamma_2 E^T PV \hat{w}^T \dot{w}^T \dot{w} + \hat{w}^T \dot{\Gamma} \hat{m} - \hat{m}^T \dot{\hat{m}} \right] \]

\[ + \frac{1}{\gamma_3} \left[ \gamma_3 E^T PV \hat{w}^T \dot{H}^T \dot{\sigma} - \hat{\sigma}^T \dot{\hat{\sigma}} \right] \]

\[ + \frac{1}{\gamma_4} \left[ \gamma_4 E^T PV \hat{w}^T (\eta - u_c) - \hat{\eta}^T \dot{\hat{\eta}} \right] \]

(34)

Substituting (28)-(31) into (34):

\[ \dot{V} = -\frac{1}{2} E^T E + 0 + 0 + 0 = -\frac{1}{2} E^T E \leq 0 \]

(35)

\(\dot{V}\) is zero only when \(E = 0.\) Thus, the equilibrium point is asymptotically stable.
FIGURE 4. A cycle of the squat movement.

IV. SIMULATION

The proposed method is varied in the combined simulation environment of MATLAB and OpenSim, as shown in Fig. 1. The simulation project can be found at https://github.com/MoranHansir/BWS_Exoskeleton.git. The human model named Gait2354 in OpenSim is used to provide guided motion. The BWS exoskeleton only connects to the back and foot. The back interaction stiffness $k_b$ and damping coefficients $b_b$ are set to $[10^4 \text{N/m}, 10^3 \text{N/m}, 500 \text{ N/rad}]$ and $[10^3 \text{N/s/m}, 10^3 \text{N/s/m}, 50 \text{ N/s/rad}]$, respectively. The foot interaction parameters are chosen as $k_f = [2 \times 10^4 \text{N/m}, 2 \times 10^3 \text{N/m}, 2 \times 10^3 \text{N/rad}]$ and $b_f = [4000 \text{ N/s/m}, 400 \text{ N/s/m}, 200 \text{ N/s/rad}]$. The human model weighs 75.16 kg, and the exoskeleton weighs 15 kg. The parameters of the exoskeleton in Fig. 2 are set as in Table 1. For BWS exoskeletons, vertical direction control is mainly considered. A squat movement is selected to verify the fuzzy CMAC-based adaptive scale force control and a condition of single leg support with the other leg swinging is used to further verify the robustness of the fuzzy CMAC-based adaptive control.

Fig. 4 shows a complete squat period. The trajectories of the human ankle, knee and hip joint are simple sinusoids with a period of $T = 4 \text{ s}$: $\theta_{\text{ankle}} = -\frac{\pi}{12} \left( \sin \left( \frac{\pi}{2} t - \frac{\pi}{2} \right) + 1 \right)$, $\theta_{\text{knee}} = -2\theta_{\text{ankle}}$, $\theta_{\text{hip}} = \theta_{\text{ankle}}$.

The gravity direction is of interest; thus, scale force control is only applied in the vertical direction, and the control goals of the other two directions are zeroes. The scale force controller is designed as $F_b = K_d F_f$, where $F_b$ is the back interaction force, $F_f$ is the sum of the foot interaction force, and the desired scale factor is $K_d = 5$. The lower level controller is the same as Equations (16) and (17) shows. The parameters are set to $K_1 = \text{diag}(700,700,150)$, $K_2 = \text{diag}(155,155,10)$.

TABLE 1. Parameters of the exoskeleton.

| Limb       | Length (m) | Mass (kg) | Center of mass (m) | Inertia (kg m^2) |
|------------|------------|-----------|--------------------|------------------|
| RF, LF     | 0.52       | 0.8       | 0                  | 0.025            |
| RS, LS     | 0.65       | 2.4       | 0.384              | 0.087            |
| RT, LT     | 0.62       | 2.0       | 0.356              | 0.052            |
| Torso      | 0.40       | 4.6       | 0.062              | 0.002            |

FIGURE 5. Squat movement without uncertainties. The squat movement begins at $t = 1 \text{ s}$. (a) Tracking errors. (b) Interaction forces.

FIGURE 6. Squat movement with model parameters decreasing by 20%. The squat movement begins at $t = 1 \text{ s}$. (a) Tracking errors. (b) Interaction forces.
FIGURE 7. Squat movement with model parameters increasing by 20%. The squat movement begins at $t = 1$ s. (a) Tracking errors. (b) Interaction forces.

$\gamma_1 = 10^6$, $\gamma_2 = 10$, $\gamma_3 = 10$, and $\gamma_4 = 10^6$. The parameters of the fuzzy CMAC network are set to $n_l = 4$ and $n_s = 9$. The inputs of the network are limited between $[-0.1$ m, $0.1$ m], $[-0.1$ rad, $0.1$ rad], $[-1$ m/s, $1$ m/s], and $[-1$ rad/s, $1$ rad/s]. If the inputs exceed the limit, the compensation of the network will not work. In the beginning, the Gaussian functions are located at the central of their corresponding block and the initial parameters $\sigma_0 = 10$. The desired trajectory $r_{eb}^d$ can be referenced using (3).

The control results of the squat movement are shown in Fig. 5. From Fig. 5a, we can see that the low-level controller tracks the desired trajectories well and that the fuzzy CMAC-based adaptive control has a better dynamic tracking effect than the inverse dynamic control. Fig. 5b shows the effect of the assistance. During the static state ($t < 1$s), the foot reaction force decreases from 736.6 N down to 113.9 N, and the rest of the body weight is supported by the exoskeleton. During the dynamic squat movement, the response of the inverse dynamic control method without the fuzzy CMAC compensation is somewhat delayed. However, the neural network-based adaptive method can always control the assistance magnification $K$ to 5 with a maximum error of 0.25. This means that the human legs only cost one-sixth of the effort to achieve the squat movement, which is conform to the scale force control goals. Note that the foot reaction force is affected by both the dynamic movements and the assistance of the exoskeleton; thus, a rapid response ability of the controller is needed.

To show the model uncertainty tolerance of the proposed method, the exoskeleton mass parameters in (13) are selected to be 80% and 120% times as large as the true value in Table. 1. The same parameters of the fuzzy CMAC adaptive scale force control are chosen. The control results are shown in Fig. 6 and 7. Both results indicate that the proposed control method has a robust ability and keeps the scale factor $K$ around the desired set value of 5 with errors between $\pm 0.96$.

Considering that the lower limb exoskeleton is mainly used for walking, decoupling the stance leg and the swing leg can...
greatly simplify the modeling and control of the biped BWS exoskeleton. Thus, a condition of single leg support with the swinging movement of the other leg is used to further research the robust ability (disturbance tolerance) of the fuzzy CMAC-based adaptive control. This condition can be regarded as the single leg support period of the walking gait or other based adaptive control. This condition can be regarded as the robust ability (disturbance tolerance) of the fuzzy CMAC-swinging movement of the other leg is used to further research exoskeleton. Thus, a condition of single leg support with the greatly simplify the modeling and control of the biped BWS

V. CONCLUSION

This work proposes fuzzy CMAC-based adaptive scale force control for portable body weight support exoskeletons. The control strategy is a two-layer framework. The high-level controller defines the control target of the system by using scale force control, and the low-level controller has strong robustness with the compensation of the neural network. The proposed control method is task free and has the advantage of decoupling the stance leg and the swing leg. By means of scale force control, the user only requires an adjustable scaled down effort to achieve normal motions, which is especially suitable for lower limb patients and elders. The robot is totally under the wearer’s control because the foot reaction force, which is equal to the leg effort, is fully controlled by the user. Squat and single leg standing movement simulations indicate that the proposed method is adequate for providing proper assistance support for users as needed.

In this research, only movements in the sagittal plane are analyzed, and only the vertical direction applies scale force control. Future work includes applying the control method on multidimensional space and making a physical prototype of the exoskeleton.

REFERENCES

[1] J. L. Pons, Wearable Robots: Biomechatronic Exoskeletons. Hoboken, NJ, USA: Wiley, 2008.
[2] H. Herr, “Exoskeletons and orthoses: Classification, design challenges and future directions,” J. NeuroEng. Rehabil., vol. 6, no. 1, p. 21, Dec. 2009.
[3] R. Steger, S. Hoon Kim, and H. Kazerooni, “Control scheme and networked control architecture for the kerleley lower extremity exoskeleton (BLEEX),” in Proc. IEEE Int. Conf. Robot. Autom. ICRA, May 2006, pp. 3469–3476.
[4] H. Kazerooni, R. Steger, and L. Huang, “Hybrid control of the kerleley lower extremity exoskeleton (BLEEX),” Int. J. Robot. Res., vol. 25, nos. 5–6, pp. 561–573, May 2006.
[5] T. Lenzi, S. M. M. De Rossi, N. Vitiello, and M. C. Carrozza, “Intention-based EMG control for powered exoskeletons,” IEEE Trans. Biomed. Eng., vol. 59, no. 8, pp. 2180–2190, Aug. 2012.
[6] Z. Li, B. Wang, F. Sun, C. Yang, Q. Xie, and W. Zhang, “SEMG-based joint force control for an upper-limb power-assist exoskeleton robot,” IEEE J. Biomed. Health Inform., vol. 18, no. 3, pp. 1034–1050, May 2014.
[7] K. Gui, H. Liu, and D. Zhang, “Toward multimodal human–robot interaction to enhance active participation of users in gait rehabilitation,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 25, no. 11, pp. 2054–2066, Nov. 2017.
[31] K. Mohajeri, M. Zakizadeh, B. Moaveni, and M. Teshnehlab, “Fuzzy CMAC structures,” in Proc. IEEE Int. Conf. Fuzzy Syst., Aug. 2009, pp. 2126–2131.

[32] Y. Sankai, “Leading edge of cybernics: Robot suit HAL,” in Proc. SICE-ICASE Int. Joint Conf., Oct. 2006, p. 1.

[33] A. Esquenazi, M. Talaty, A. Packel, and M. Saulino, “The ReWalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury,” Amer. J. Phys. Med. Rehabil., vol. 91, no. 11, pp. 911–921, Nov. 2012.

[34] S. Wang, L. Wang, C. Meijneke, E. van Asseldonk, T. Hoellinger, G. Cheron, Y. Ivanenko, V. La Scaleia, F. Sylos-Labini, M. Molinari, F. Tamburella, I. Pisotta, F. Thorsteinson, M. Ilzakovitz, J. Gancet, Y. Nevatia, R. Hauffe, F. Zanow, and H. van der Kooij, “Design and control of the MINDWALKER exoskeleton,” IEEE Trans. Neural Syst. Rehabil. Eng., vol. 23, no. 2, pp. 277–286, Mar. 2015.

[35] A. Seth, J. L. Hicks, T. K. Uchida, A. Habib, C. L. Dembia, J. J. Dunne, C. F. Ong, M. S. DeMers, A. Rajagopal, M. Millard, S. R. Hamner, E. M. Arnold, J. R. Yong, S. K. Lakshmikanth, M. A. Sherman, J. P. Ku, and S. L. Delp, “OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement,” PLOS Comput. Biol., vol. 14, no. 7, Jul. 2018, Art. no. e1006223.

HAN WU received the B.S. and M.S. degrees from the National University of Defense Technology, in 2014 and 2017, respectively, where he is currently pursuing the Ph.D. degree in control science and engineering.

His research interests include nonlinear control, exoskeletons, and neural network control.

HONGLEI AN received the B.S., M.S., and Ph.D. degrees from the National University of Defense Technology (NUDT), in 2006, 2009, and 2013, respectively.

From 2013 to 2019, he was a Lecturer at NUDT. His research interests include legged robot control, nonlinear control theory, and optimal control application.

QING WEI received the B.S. degree in radio electronics from Fudan University, in 1991, the M.S. degree in automatic control, in 1993, and the Ph.D. degree in automatic control and application from the National University of Defense Technology, in 1996.

In 2001, he was a Visiting Scholar of The Hong Kong University of Science and Technology. From 2015 to 2016, he was a Senior Visiting Scholar of the University of Southampton, U.K.

His research interests include legged robot control, nonlinear control theory, and optimal control application.

HONGXU MA received the B.S. degree in automatic control, in 1988, the M.S. degrees in intelligent control, in 1991, and the Ph.D. degree in control science and control engineering from the National University of Defense Technology, in 1995.

In 1995, he stayed in the school for research work. He is currently a Professor with the National University of Defense Technology. He is mainly engaged in the research of legged robots. His research interests include legged robot control, nonlinear control, and pattern recognition.

* * *