Multi-fuzzy Sarsa learning-based sit-to-stand motion control for walking-support assistive robot

Xu Wenxia¹, Baocheng Yu¹, Lei Cheng², Yanan Li¹ and Xuan Cao¹

Abstract
Sit-to-stand transfer is a very common and critical movement of daily life in elderly individuals, especially independent elderly individuals. However, most assistive robots do not have a sit-to-stand transfer function. In this article, a multi-fuzzy Sarsa learning-based sit-to-stand motion control method for walking-support assistive robot was proposed. First, the mechanical design of walking-support assistive and sit-to-stand transfer motion control problems were introduced. Then, the fuzzy Sarsa learning method, which is a model-free algorithm, was used to design the motion control algorithm for the human–robot system. To realize natural and intuitive sit-to-stand transfer movement for a human–robot system, the interactive force between the robot and human and the error position between the real-time center of mass and reference center of mass were state variables of the proposed fuzzy Sarsa learning-based sit-to-stand motion control algorithm. Considering the computing efficiency of the controller, a multi-fuzzy Sarsa learning-based motion control algorithm was developed to realize natural sit-to-stand transfer motion. Finally, the experimental results verify the effectiveness of the proposed algorithm.

Keywords
Sit-to-stand transfer, walking-support robot, fuzzy Sarsa learning, the center of mass

Date received: 11 March 2020; accepted: 14 September 2021

Topic: Service Robotics
Topic Editor: Marco Ceccarelli
Associate Editor: Kenji Hashimoto

Notation
- $M(q)$: inertia matrix of the human body model
- $C(q, \dot{q})$: Coriolis effects matrix of the human body model
- $G(q)$: gravitational forces vector of the human body model
- $q$: joint angle of the user
- $\tau_{\text{robot}}$: driving torque of the walking-support assistive robot
- $\tau_{\text{num}}$: active joint driving torque of the user
- $d$: disturbance of the dynamic model
- $m_i$: the mass of the user’s segment
- $p_i$: the mass position of the user’s segment
- $P_{\text{com}}$: the center of mass (COM)
- $s_t$: state variable at instant $t$ of RL algorithm
- $a_t$: the selected action at instant $t$ of RL algorithm
- $p(s_t, a_t, s_{t+1})$: transition probability of state transformation
- $r_t$: the reward at instant $t$ of RL algorithm
- $Q$: action value of RL algorithm
- $\gamma$: discount factor

¹Hubei Key Laboratory of Intelligent Robot, Wuhan Institute of Technology, Hubei, China
²School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan, Hubei, China

Corresponding author:
Baocheng Yu, Hubei Key Laboratory of Intelligent Robot, Wuhan Institute of Technology, Wuhan, Hubei 430205, China.
Email: 11020501@wit.edu.cn
Introduction

Due to technological advancements and extended human longevity, over the last 60 years, the rate of aging has accelerated markedly. Moreover, elderly nursing services spend too many health-care resources and money to meet elderly people’s needs for rehabilitation and independent living. Therefore, developing auxiliary support equipment that provides physical support and recovery and allows the elderly’s to walk and perform daily activities is urgently required.1 Due to cognitive disorders and lower limb dysfunction, standing up and sitting down are prevalent and critical activities of daily life in elderly individuals, especially independent elderly individuals. Falls often occur during walking, standing up, and sitting down.2 Hence, the sit-to-stand (STS) transition is considered critical to an individual’s quality of life and functional independence.3 Thus, there is a great need for walking-assist robotic systems to help elderly or disabled people stand up independently.

Currently, walking-assist robotic systems with STS transfer functions can be categorized into two types: exoskeleton robots and walking vehicle robots.4 The exoskeleton robots can help the use to stand up and sit down. However, few walking vehicle robots with STS transfer functions have been developed. Mederic et al.5 designed an assist device that provides physical support to elderly people during walking and STS transfer. Chuy et al.6 developed a robotic walking support system called “Walking Helper II.” Omar et al.7 described a mobility assisting device that helps patients who do not have enough physical strength in their lower limbs during STS transfer.

The control theories for walking-assist robotic STS transfer can be grouped into three categories.5 (1) Motion control is the first category. Mederic et al.5 described the trajectory generation problem of assistive device handles with interpolating cubic splines. Then, by guiding the angle and height of the support plate, the assistive device followed the desired trajectory. Omar et al.7 proposed a robot motion control algorithm that follows the natural pattern of human motion during STS transfer and provides assistance force to the user’s shoulder. (2) Force control is the second category. Mederic et al.9 calculated the zero moment point of the user and controlled the interaction force between the robot and the user to maintain balance during STS transfer. (3) Switching control is the third category. Different control methods were used in the different phases of STS transfer.10 In the lifting body phase, a damping control method was implemented with a force reference, and in other stages of STS transfer, compliant impedance control was used in other stages of STS transfer.

In fact, elderly or disabled people are encouraged to practice walking and STS transfer to prevent the degeneration of residual mobility abilities. Therefore, the motion control algorithm for assistive robots should consider the user’s motion intention and give a degree of assistance to the user rather than replace the user’s motion capability completely. To encourage the user to practice daily activities as much as possible, the coupled human–robot system should behave naturally and intuitively. Hence, the assistive robot should have the ability to understand and imitate human behavior during STS transfer. However, establishing an exact model for the coupled human–robot system is difficult because accurate active joint torque values are difficult to obtain. In addition, time-varying uncertainties will be raised from the coupled human–robot system and environmental interactions.

The good performance of traditional adaptive control methods rely on an exact system model.11,12 Hence, model-free adaptive control techniques are feasible for these problems. Jabbari et al.13 proposed a neural network-based trajectory tracking control method for robotic exoskeletons, and Hussain et al.14 designed an adaptive impedance controller for a robotic orthosis. To solve the structure and parameters uncertainty and disturbance problem of upper limb exoskeleton, Li et al.15 utilized adaptive fuzzy approximators to estimate the dynamical uncertainties of the human–robot system, and an iterative learning scheme was employed to compensate for unknown time-varying periodic disturbances.

Over the last several years, reinforcement learning (RL) has caught the research attention in terms of exoskeletons and walking-assist devices.16 Huang et al.17 designed a hierarchical interactive learning (HIL)-based control strategy to learn piecewise linear models of lower limb exoskeletons. High-level motion learning is dynamic movement primitives combined with locally weighted regression, and the lower-level motion learning method is RL. In control paradigms, the RL-based interactive learning control
method can reduce the complexity of the coupled human–robot system, adapt to uncertainty and address varying interaction dynamics problems. RL-based motion control methods are mainly used in the trajectory tracking motion control of exoskeletons and walking-assist devices and rarely involve STS transfer. Therefore, in this article, RL method was used in STS transfer for walking-support assistive robot. The serious drawback of RL is the dimension explosion problem, which is the result of the large dimension of the discrete state–action pairs. To solve this problem, fuzzy RL (FRL) is usually used. Fuzzy Sarsa learning (FSL) is a well-known RL method that has a convergence theorem. Compared to fuzzy Q learning, FSL has a significantly higher learning speed and action quality. Therefore, we develop the FSL-based STS motion control for the walking-support assistive robots.

The main contributions of the article include the following:

(i) Propose the RSL-based motion control method for the STS motion control of walking-support assistive robots;
(ii) Decrease the dimension of state–action pairs by using two FSL-based STS motion controllers to control the linear actuator and mobile base of the robot;
(iii) Apply the proposed method in practice for the STS motion control of walking-support assistive robots.

In this article, a multi-FSL-based STS motion control method for walking-support assistive robot is proposed. First, the walking-support assistive robots is introduced. Two value indexes of the STS problem for walking-support assistance are proposed. Then, the FSL algorithm is described. To realize the natural and intuitive STS transfer movement, two FSL-based motion control algorithms are designed. Finally, some experiments were conducted to validate the effectiveness of the proposed multi-FSL-based STS motion control algorithm.

**Walking-support assistive robot**

**Mechanics of the walking-support assistive robot**

An overview of the walking-support assistive robot is described in Figure 1. It consists of a mobile base, an STS support subsystem, and a human–robot interactive (HRI) system. The mobile base consists of two passive casters and two differential driving wheels. The two passive casters are mounted under the controller, and the two driving wheels are installed on the extended base, as shown in Figure 1. The extended base extends the robot’s support area and provides relatively safe and comfortable support. The STS support subsystem contains a linear actuator and a support frame. The linear actuator is the power source of the STS support subsystem, which helps the user realize STS transfer movement. To increase the support area and improve stability and safety, the support frame has two jointed supporting bars, as shown in Figure 1.

The HRI system includes two force-sensing resistance (FSR)-based HRI subsystems and wearable sensor (WS)-based HRI subsystem.

(i) FSR-based HRI subsystem

Two FSR sensors are installed underneath the two support plates, as shown in Figure 2, to detect the interactive forces between the user and the robot, which reflects the user’s intention of standing up.

(ii) WS-based HRI subsystem

To realize a natural and intuitive STS transfer movement, identifying user body parts is critical. The triple inverted pendulum has been widely used as a simplified biomechanical model of the human body in the STS...
transfer literatures.\textsuperscript{10,20} In this article, we also used a triple inverted pendulum to represent the human body, as shown in Figure 3. To obtain the real-time position information of the human body, five wearable sensor units were attached on the user’s waist, thighs, and shanks, as shown in Figure 4. A wearable sensor unit consists of a triaxial magnetometer, a triaxial accelerometer, and a triaxial gyroscope for measuring the acceleration and angular velocity along three orthogonal axes simultaneously. The triaxial accelerometer and magnetometer of each sensor unit are integrated on a chip (LSM303). The product model of the triaxial gyroscope is MPU3050. An STM32F020 Micro Control Unit (MCU) is used to collect data for all sensor units, which detect the position of the user’s trunk, thighs, and shanks; the details can be seen in the literature.\textsuperscript{21}

**Dynamic model of human body model**

The walking-support assistive robot and human consist of a coupled human–robot system, which is shown in Figure 3. The dynamic model of the human body model can be expressed as follows

\[
M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau_{\text{robot}} + \tau_{\text{hum}} + d
\]  

(1)

However, obtaining accurate driving torque $\tau_{\text{hum}}$ of active joints is difficult. Then, in this article, the RL method, which is a model-free algorithm, was used to design the control law of the walking-support assistive robot.

**STS transfer problem for walking-support assistive robot**

STS is a common movement but difficult movement of daily life in the elderly, as humans require a high amount of energy and joint force to stand up. However, it is dangerous for the elderly to stand up without any auxiliary instruments. In the process of STS transfer movement, safety and compliance are most important for the elderly. To ensure the user’s safety, the robot should maintain human balance and postural stability during STS transfer.

In consideration of the requirements of natural and intuitive STS transfer, the STS transfer motion control for the robot should consider the following two points.
Center of mass (COM): The COM is widely used to evaluate human balance and represents the balance point of an object’s mass. When the system is balanced around its COM, then it is in a state of equilibrium. In this article, the COM of the user can be calculated by the following equation

$$p_{\text{com}} = \frac{\sum_{i=1}^{5} m_i \hat{p}_i}{\sum_{i=1}^{5} m_i}$$

where $m_i$ and $\hat{p}_i$ are the mass and mass positions of the human segment. The trajectory of the human’s COM during the STS transfer movement is shown in Figure 5.

Interactive force: The interactive forces between the robot and the user express the user’s motion intention. At the beginning of the STS transfer, a large value of interactive forces means that the user wants to stand up. In the process of STS, the interactive forces decrease slowly. When user stands almost straight, the interactive forces approach zero at the end of the STS transfer.

### Fuzzy Sarsa learning

The human user and walking-support assistive robot constitute a coupled human–robot system, and establishing an accurate mathematical model for this system is difficult. RL is an online model-free incremental learning technique that is widely used in coupled human–robot systems. However, the large dimensionality of discrete state–action pairs will cause the curse of dimensionality problem. FRL was used in the motion control of walking-support assistive robots because it can overcome the curse of dimensionality problem.

Before designing the robot motion control algorithm, a brief explanation of the RL and Sarsa learning (SL) algorithms is introduced in this section.

**Reinforcement learning**

In RL, at each time step $t$, the agent observes the current state $s_t$ of the environment. Then, the agent executes action $a_t$ from the set of actions $A$ under the action selection policy $\pi$ and obtains reward $r_t$. Consequently, the state transfers to $s_{t+1}$ with transition probability $p(s, a, s_{t+1})$. Then, the agent selects action $a_{t+1}$ and obtains reward $r_{t+1}$. The interactive cycle between an agent and the environment will eventually converge to the best result. The schematic diagram of RL is shown in Figure 6.

The value of action $a$ in state $s$ under policy $\pi$ can be estimated according to the following formula

$$Q^\pi(s, a) = E_{\pi} \left[ \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} | s_t = s \right]$$

where $\gamma$ is the discount factor, with $0 \leq \gamma \leq 1$.

The most popular action policies include the $\varepsilon$-greedy action selection method and Soft-Max method. The $\varepsilon$-greedy action selection method selects the action (or one of the actions) with the highest estimated action value $Q$. The Soft-Max method ranks and weights all actions with their value estimates $Q$, and then the action with the highest selection probability is selected.

**Sarsa learning**

SL is a well-known RL methods that is an online model-free learning technique. SL has been widely applied to a variety of problems in robotics. Herein, the SL algorithm is used in the motion control of STS transfer for walking-support assistive robots.

In the SL algorithm, the future reward is the actual reward for executing real action. Therefore, the value of
action \( Q(s, a) \) in state \( s \) is updated with the value of action and the current policy using equation (4)

\[
Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha_t[r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]
\]

where \( r_{t+1} \) denotes the immediate reward received from the environment after applying action \( a_t \) in state \( s_t \).

**Fuzzy Sarsa learning**

FSL is an extension of SL. A zero-order T-S fuzzy system is used to approximate the function of a continuous state and action space. In the FSL algorithm, the input state vector \( s = \{x_1, x_2, \cdots, x_n\} \), \( n \) is the number of input state vectors. For the STS motion control of the human–robot system, the input state variable \( x_1 \) is the trajectory error of the user's COM and desired COM and \( x_2 \) is the HRI force, as shown in Figure 7. The schematic diagram of FSL is shown in Figure 7. In this figure, the rule firing strengths of each state variable are generated by a Gaussian membership function, and the normalized firing strength of \( i^\text{th} \) rule for state \( x_j \) is \( \mu_i(x_j) \). In each rule, the possible discrete control action set is \( e = \{u_1, u_2, \cdots, u_m\} \), which consists of the velocity of the linear actuator. The \( R_i, i \in N \) rules of the T-S system is as follows

\[
R_i: \text{If } x_1 \text{ is } L_1^i, \cdots, \text{and } x_n \text{ is } L_n^i, \text{ then } u = u^i \text{ with } \omega^i
\]

where \( \omega^i \) is the approximate value of \( \mu^i \)th candidate action in \( i^\text{th} \) rule and \( L^i_j \) is the linguistic term for variable \( x_j \) under rule \( R_i \).

Then, the system output at instant \( t \) is calculated as follows

\[
c_t(s_t) = \sum_{i=1}^{N} \mu_i(s_t)u_i^o
\]

where \( o \) is the index of optimal action and \( u_i^o \) is the optimal action in rule \( R_i \) according to the following Softmax policy:

\[
p(u_i^o) = \frac{\exp(\mu_i^o/T)}{\sum_{k=1}^{N} \exp(\mu_k^o/T)}
\]

The action value function is computed by the following equation

\[
Q_t(s_t, c_t) = \sum_{i=1}^{N} \mu_i(s_t)\omega_i^o
\]

where \( \omega_i^o \) is the weight parameter of optimal action in rule \( R_i \).

The system executes the inferred optimal action \( c_t \) at instant \( t \). Then the system transfers to the next state at instant \( t + 1 \) and generates a reinforcement signal ‘r’. This signal \( r \) is used to calculate the approximate action value error \( \Delta Q_t(s_t, c_t) \) as follows

\[
\Delta Q_t(s_t, c_t) = r(s_t, c_t) + \gamma Q_t(s_{t+1}, c_{t+1}) - Q_t(s_t, c_t)
\]

The weight value \( \omega(s_t, u_j) \) of the \( i^\text{th} \) rule is updated by the following equation

\[
w_{ij}^o = w_{ij}^o + \alpha \Delta Q_t(s_t, c_t) \mu_i(s_t)
\]

If \( j \) is the index of the selected action weight, then \( \Delta Q_t(s_t, c_t) \mu_i(s_t) = 0 \).

The whole process of FSL is summarized in Table 1.

**Table 1. The process of FSL algorithm.**

| Input: \( s_t \) | Output: \( c_t \) |
|------------------|------------------|
| 1. Get the observe state \( s_{t+1} \) and reward value \( r_{t+1} \). | |
| 2. Choose an action for each rule by equation (7). | |
| 3. Calculate the system output action \( c_{t+1} \) and \( Q_t \) according to equations (6) and (8), respectively. | |
| 4. Calculate the \( \Delta Q_t \) and \( \omega \) according to equations (9) and (10), respectively. | |
| 5. Calculate the new approximate \( Q_{t+1} \) using equation (8). | |
| 6. Send the system output \( c_t \) to the robot controller. | |
| 7. \( t = t + 1 \), return to step 1. | |

**FSL: fuzzy Sarsa learning.**

**Multi-FSL-based STS motion control for the walking-support assistive robot**

Elderly individuals are encouraged to practice STS transfer movements to prevent the degeneration of residual mobility ability. Hence, a walking-support assistive robot is...
designed to help the user stand up in this article. Actually, the robot should assist humans in standing up independently rather than pulling them up. That is, the motion controller of the robot should take the user’s motion intention fully into account, behave naturally and intuitively, and follow human STS movements. Then, the input state variables are the interactive force $F_c$ in the coupled human–robot system and the trajectory errors $e = \{e_x, e_z\}$ of the user’s COM in the horizontal direction and vertical direction. The system outputs are the velocities $v = \{v_x, v_z\}$ of the linear actuator and mobile base of the robot in the horizontal direction and vertical direction, respectively. Then, the control object of the proposed algorithm obtains the optimal velocities of the linear actuator and mobile base of the robot, which helps the user naturally and intuitively realize STS transfer movements

$$e_x = x_{\text{com}} - x_{\text{rcom}}, e_z = z_{\text{com}} - z_{\text{rcom}}$$

where $x_{\text{com}}$ and $x_{\text{rcom}}$ are the actual position and reference position of the COM in the horizontal direction and $z_{\text{com}}$ and $z_{\text{rcom}}$ are the actual position and reference position of the COM in the vertical direction.

To decrease the dimensionality of state–action pairs, two FSL-based STS motion controllers were used in this article to control the linear actuator and mobile base of the robot, as shown in Figure 8. The FSL-based motion controller 1 was used to generate the control command of the linear actuator. The input state variables were $F_c$ and $e_x$, and the output was $v_x$. FSL-based motion controller 2 was used to generate the control command of the mobile base in the horizontal direction. The input state variables were $F_c$ and $e_z$, and the output was $v_z$. And $F_c$ and $e_z$ are uniformly partitioned by five fuzzy sets, $e_x$ is uniformly partitioned by three fuzzy sets. The defined Gaussian membership functions for the three inputs are shown in Figure 9.

The output of FSL-based motion controller 1 determines the motion velocity of the linear actuator. The candidate actions set is $U = \{0, 0.01, 0.03, 0.07, 0.1, 0.12, 1.15\}$ m/s. The reward function is defined as follows

$$r(t)_1 = \begin{cases} 1, & e_x = 0 \\ 0.7, & 0 \leq e_x < 10 \text{ cm}, \dot{e}_x < 0 \\ 0.3, & 10 \text{ cm} < e_x, \dot{e}_x < 0 \\ 0, & \dot{e}_x > 0 \end{cases}$$

The output of FSL-based motion controller 2 determines the motion velocity of the mobile base. The candidate actions set is $U = \{-0.01, 0, 0.005, 0.01, 0.02, 0.03, 0.04\}$ m/s. The reward function is defined as follows

$$r(t)_2 = \begin{cases} 1, & e_z = 0 \\ 0.5, & 0 \leq e_z < 10 \text{ cm}, \dot{e}_z < 0 \\ 0, & \dot{e}_z > 0 \end{cases}$$

**Experimental evaluation**

In this section, some experiments were conducted to verify the proposed multi-FSL-based STS motion control algorithm for walking-support assistive robots. First, the process of the experiment was introduced. Then, to verify the effectiveness of the proposed algorithm, some STS experiments with walking-support assistive robots were performed and a discussion was provided after the experiments.
Experiment process

The experiments were conducted by one subject (aged 33, height 160 cm, weight 52 kg). According to the method of calculating the mass of human segments, the mass of each segment is shown in Table 2. In the experiment, the subject was asked to wear wearable sensor units, as shown in Figure 4. Then the subject grabbed the handles and sat on the chair, as shown in Figure 10. The walking-support assistive robot detected the interactive force and the robot started to help the user stand up. The process of STS with the walking-support assistive robot is delineated as in Figure 10.

Before the experiments, the subject was asked to stand up with wearable sensors 30 times. The wearable sensors detected the motion positions of the user’s two legs. Then, the reference trajectory-fitted curve of the user’s COM can be calculated, as shown in Figure 11.

Training of weight values

The output action of the FSL algorithm is the weighted sum of the selected actions in the rules. Therefore, in the FSL algorithm, \( \omega(s, c) \) is critical. However, the FSL algorithm needs many trials to obtain the optimal output action. Then, it will spend considerable time conducting the experiments with the walking-support assistive robot. To decrease the time of the experiment, a simulated experiment was implemented before the experiments with the robot to train \( \omega(s, c) \).

To obtain input–output training data of the FSL algorithm, the robot and subject implemented an STS transfer trial to gather the interactive force and COM data of the coupled human–robot system. Then, according to the dynamic model equation (1), \( \omega(s, c) \) of the candidate actions in each rule was trained by the proposed method 1000 times. Finally, we conducted the experiments with a walking-support assistive robot. The details of the experimental results and discussion are explained in the next section.

Experimental results and discussion

In this section, we evaluate the proposed algorithms by three measures:

![Figure 9. The membership degree of three input state variables.](image)

**Figure 9.** The membership degree of three input state variables.

| Table 2. The parameters of subject. |
|-------------------------------------|
| Parameter | Weight (kg) | Description               |
|-----------|-------------|---------------------------|
| \( m_1 \) | 29.1        | Mass of HAT               |
| \( m_2 \) | 2.24        | Masses of arms            |
| \( m_3 \) | 24.62       | Mass of torso-head        |
| \( m_4 \) | 7.39        | Masses of thighs          |
| \( m_5 \) | 3.04        | Masses of shanks          |

**Figure 10.** The process of STS with walking-support assistive robot. STS: sit-to-stand.

| Table 3. The parameters of FSL algorithm. |
|-------------------------------------------|
| Parameter | \( \gamma \) | \( \alpha \) | \( T \) | \( \Delta t \) |
|-----------|--------------|--------------|--------|--------------|
|           | 0.9          | 0.05         | 0.95   | 0.05         |

FSL: fuzzy Sarsa learning.
Trial time is the time of a complete STS trial; MCE is the mean COM error, thus representing the average error between the user’s actual COM trajectory and the reference trajectory.

\[
MCE = \bar{e}_x + \bar{e}_z = \frac{\sum_{i=1}^{N} (e_x(n)) + \sum_{i=1}^{N} (e_z(n))}{N} \tag{14}
\]

where \(N\) is the sampling times in each trial, \(e_x(n)\) and \(e_z(n)\) are the trajectory errors of the user’s COM in the horizontal direction and vertical direction on the \(n^{th}\) sampling time in a trial.

(iii) MIF is the mean of the interactive force, thus representing the average interactive force in a trial.

\[
MIF = F_c = \frac{\sum_{i=1}^{N} (F_c(n))}{N} \tag{15}
\]

where \(F_c(n)\) is the interactive force on the \(n\)-th sample time and \(x_{com}(n)\) is the desired position of COM in the X-axis on the \(n^{th}\) sampling time in a trial.

The experiments included 30 trials. Table 4 shows the experimental results. The average trial time was computed by taking the average of 30 STS trials. As seen in Table 4, after 30 trials, all three measures significantly outperformed those of the first trial, with the Trial time, MCE, and MIF decreasing by 16.8%, 64.5%, and 27.2% compared with the first trial.

The purpose of this robot is to allow robots to assist user to achieve STS transfer and make the user feel that the robot is following user’s standing movements rather than just supporting the user to stand up. That is, the robot needs to assist the user’s actual COM closer to the reference COM. And according to the results of the subject questionnaire survey, the subject does felt that the robot was following her STS transfer movement rather than simply supporting the user standing up. In other words, the robot realizes a more comfortable STS transfer. Therefore, the interactive force was less than that in the first trial.

Figures 12 shows the curve of the three measurement indexes. Figures 13 and 14 show the experimental results in the last trial. The STS transfer movement can be divided into several phases, namely, preparation, rising, and stabilization. Based on Figure 14, the preparation stage is in the time interval \((0 \leq t < 4.5)\) [s], the rising phase is in the time interval \((4.5 \leq t < 7.4)\) [s], and the stabilization phase is for \(t \geq 7.4\) [s]. In the preparation phase, the error of the COM in the horizontal direction \(e_x\) does not change much, and the velocity \(v_x\) is almost zero. The interactive force was very large, as the user was ready to stand. In the lifting phase, the user left the chair and the position of the COM in the vertical direction increased rapidly; then, \(v_x\) also increased with \(e_x\). In the stabilization phase, the position of the COM no longer changes, and \(v_x\) is almost zero. The linear actuator ascended with the whole process of STS transfer, as shown in Figure 13.

### Comparative experiment

To verify the effectiveness of the proposed STS transfer motion control algorithm, the motion control algorithm in the literature was compared with the proposed algorithm in this article. The robot in the literature is the same as our walking-support assistive robot. Autonomous control uses
the voltage across the FSR as the control signal, while posture control uses the attitude angles in the literature. The three indexes (IMF and Trial time) of the two motion control algorithms are shown in Table 1. According to Table 5, the proposed method has better performance than in the literature.

**Conclusion**

A multi-FSL-based STS motion control for walking-support assistive robot was introduced in this article. According to the requirements for natural and intuitive STS transfer movement, a multi-FSL-based STS motion control algorithm for walking-support assistive robot was proposed. The mechanism of walking-support assistive and STS transfer problem were briefly introduced. Considering the difficulty in obtaining accurate mathematical model for the coupled human–robot system, FSL method was used to control the robot. To increase the efficiency of FSL algorithm, two FSL-based controllers were implemented to control linear actuator and mobile base, respectively. At last, walking-support assistive robot experiments have proved the validity of the proposed algorithm.

The presented control scheme for walking-support assistive robot was only evaluated on healthy subjects. And the robot needs time and experimentations to learn the user’s STS transfer movement. The elderly may not be able to conduct too many trials. To establish the effect of multi-FSL-based STS motion control, elderly individual trials are necessary and are currently undergoing as a part of this research.

### Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

### Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was funded by the National Natural Science Foundation Youth Fund of China Grant Numbers 61803286 and 61906139 and the Nature Science Foundation (Youth fund) of Hubei province Grant Number 2018CFB163.

### ORCID iD

Xu Wenxia [https://orcid.org/0000-0002-1101-418X](https://orcid.org/0000-0002-1101-418X)

---

**Table 5.** Comparative experiment results for STS.

| Trial time (s) | MIF          |
|---------------|--------------|
| STS method    | 12.7         | 36.25 N     |
| Multi-FSL-based STS | 10.4 | 27.44 N     |

STS: sit-to-stand; MIF: mean of the interactive force; FSL: fuzzy Sarsa learning.
References

1. Bedaf S, Gelderblom GJ and De Witte L. Overview and categorization of robots supporting independent living of elderly people: what activities do they support and how far have they developed. Assist Technol 2015; 27(2): 88–100.

2. Rubenstein LZ. Falls in older people: epidemiology, risk factors and strategies for prevention. Age Ageing 2006; 35(Suppl 2): ii37–ii41.

3. Ploutz-Snyder LL, Manini T, Ploutz-Snyder RJ, et al. Functionally relevant thresholds of quadriceps femoris strength. J Gerontol A 2002; 57(4): B144–B152.

4. Zhao DH, Yang JY, Okoye MO, et al. Walking assist robot: a novel non-contact abnormal gait recognition approach based on extended set membership filter. IEEE Access 2019; 7: 76741–76753.

5. Mederic P, Pasqui V, Plumet F, et al. Sit to stand transfer assisting by an intelligent walking-aid. In: Climbing and Walking Robots 2005, Berlin, Heidelberg: Springer; pp. 1127–1135.

6. Chuy O, Hirata Y, Wang Z D, et al. Approach in assisting a sit-to-stay movement using robotic walking support system. In: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Bejing, China, 9 October–15 October 2006, paper no. IROS 2006, pp. 4343–4348. IEEE.

7. Omar S, Ahmed A, Ahmed MR, et al. Development of parallel manipulator sit to stand assistive device for elderly people. In: 2013 IEEE Workshop on Advanced Robotics and its Social Impacts, Tokyo, Japan, 7 November–9 November 2013, paper no. ARSO, pp. 27–32. IEEE.

8. Geravand M, Korondi PZ, Werner C, et al. Human sit-to-stand transfer modeling towards intuitive and biologically-inspired robot assistance. Auto Robot 2017; 41(3): 575–592.

9. Mederic P, Pasqui V, Plumet F, et al. Elderly people sit to stand transfer experimental analysis. In: International Conference on Climbing and Walking Robots, Berlin, Heidelberg, 13 September–15 September 2005, paper no. CLAWAR 2005, pp. 953–960. Springer.

10. Chugo D, Morita Y, Sakaida Y, et al. A robotic walker for standing assistance with realtime estimation of patients load. In Proceeding of 12th IEEE International Workshop on Advanced Motion Control, Sarajevo, Bosnia-Herzegovina, 25 March–27 March 2012, paper no. AMC 2012. IEEE.

11. Huang J, Ri S, Fukuda T, et al. A disturbance observer based sliding mode control for a class of underactuated robotic system with mismatched uncertainties. IEEE Trans Auto Cont 2019; 64(6): 2480–2487.

12. Huang J, Cao Y and Wang YW. Adaptive proxy-based sliding mode control for a class of second-order nonlinear systems and its application to pneumatic muscle actuators. ISA Transaction, 2020.

13. Jabbari Asl H, Narikiyo T and Kawashishi M. Neural network based bounded control of robotic exoskeletons without velocity measurements. Control Eng Pract 2018; 80: 94–104.

14. Hussain S, Xie SQ and Jamwal PK. Adaptive impedance control of a robotic orthosis for gait rehabilitation. IEEE Trans Cyber 2013; 43(3): 1025–1034.

15. Li Z, Su CY, Li G, et al. Fuzzy approximation-based adaptive backstepping control of an exoskeleton for human upper limbs. IEEE Trans Fuzzy Sys 2015; 23(3): 555–566.

16. Xu WX, Huang J, Wang YJ, et al. Reinforcement learning-based shared control for walking-aid robot and its experimental. Adv Robot 2015; 29(22): 1463–1481.

17. Huang R, Cheng H, Guo HL, et al. Hierarchical interactive learning for a human-powered augmentation lower exoskeleton. 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 2016, pp. 257–263.

18. Khan SG, Tufai ML, Shah SH, et al. Reinforcement learning based compliance control of a robotic walk assist device. Adv Robot 2019; 33(24): 1–12.

19. Er MJ and Deng C. Online tuning of fuzzy inference systems using dynamic fuzzyQ-learning. IEEE Trans Sys Man Cyber Part B 2004; 34: 1478–1489.

20. Zijlstra W, Bisseling RW, Schlumbohm S, et al. A body-fixedsensor-based analysis of power during sit-to-stand movements. Gait Posture 2010; 31: 272–278.

21. Huang J, Xu WX, Mohammed S, et al. Posture estimation and human support using wearable sensors and walking-aid robot. Robot Auto Sys 2014; 73: 24–43.

22. Motlagh FN and Arshi AR. Symmetry comparison between sacrum and center of mass during walking. In: Institution of Mechanical Engineers Part H-Journal of Engineering in Medicine, 1 July 2016, paper no. 230(7), pp. 682–689. SAGE.

23. Howard M, Braun DJ and Vijayakumar S. Transferring human impedance behavior to heterogeneous variable impedance actuators. IEEE Trans Robot 2013; 29: 847–862.

24. Kronander K and Billard A. Online learning of varying stiffness through physical human-robot interaction. In: 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN, USA, 2012, pp. 1842–1849.

25. Li X and Hirasawa K. Continuous probabilistic model building genetic networkprogramming using reinforcement learning. Appl Soft Comput 2015; 27: 457–467.

26. Leva P. Adjustments to zatsiorsky-seluyanov’s segment inertia parameters. J Biomech 1996; 29(8): 1223–1230.

27. Lindemann U, Claus H, Stuber M, et al. Measuring power during the sit-to-stand transfer. Eur J Appl Physiol 2003; 89(5): 466–470.

28. Yang DQ, Zhou M, Huang J, et al. Aided sit to stand transfer by assistive robot and wearable sensors. In: 2017 2nd International Conference on Advanced Robotics and Mechatronics, Hefei, China, 27 August–31 August 2017, paper no. ICARM 2017, pp. 239–244. IEEE.