CrossMark

Optimisation of Reference Gait Trajectory of a Lower Limb Exoskeleton

Rui Huang[1] · **Hong Cheng**[1] · **Yi Chen**[1] · **Qiming Chen**[1] · **Xichuan Lin**[1] · **Jing Qiu**[1]

Abstract Lower limb exoskeletons have gained considerable interest in recent years as a research topic for creating aids for people with walking disabilities and strength augmenters for pilot walkers. A crucial practical problem, however, is generating the reference trajectory of the joints. In this paper, we solve the reference trajectory problem by a novel approach which obtains the angle trajectories of knee joints from the hip joints. The relationship between the angle trajectories of the knee and hip joints is acquired through kinematic models of the lower limb exoskeleton. In these models, the parameters of the joint position trajectories are optimised by a swarm fish algorithm with variable population. The proposed approach is validated in virtual simulations and a physical prototype of an exoskeleton system. The experimental results confirm that the reference trajectory generation approach accurately reproduces human walking.

Keywords Lower limb exoskeleton · Kinematic model · Trajectory generation · Swarm fish algorithm

1 Introduction

Lower limb exoskeletons can provide a wide range of human assistance, from standing/walking aids for people with lower limb disabilities to strength augmentation for pilot walkers. As such, they have become increasingly promoted in recent years. A group of researchers headed by Sankai have developed the robot suit hybrid assistive limb (HAL) exoskeleton

at Tsukuba University [1,2]. The numerous functions on the HAL exoskeleton assist doctors and physical therapists with their training, the disabled with their standing/walking activities, and nurses with transporting heavier patients. Argo Medical Technology has been developing a rehabilitation exoskeleton named 'Rewalk', designed for patients who are completely disabled in lower limbs [3]. With the operator affixed to their upper bodies, patients can operate 'Rewalk' to accomplish actions such as walking, climbing stairs, shifting from sitting to standing and self-support. Another rehabilitation exoskeleton system, the exoskeleton lower extremity gait system (eLEGS; later known as Ekso), has been designed by Berkeley Bionics (Berkeley University, California, USA) [4,5]. eLEGS intelligently interprets patients' intents from their motion information and translates them into appropriate action. Various lower limb exoskeletons have also been designed for military use [6,7], the most successful of which is Berkeley lower extremity exoskeleton, also designed at Berkeley [8,9].

There are two challenge problems in control of lower limb exoskeleton, controller design and motion trajectories generation. On the control of lower limb exoskeletons, many adaptive control methods were proposed to achieve better performance [10–13], especially in the applications of rehabilitation robots [14,15]. In the study of this paper, we focus on generating the reference trajectory of lower limb exoskeletons. Most of the reference trajectory generation methods in humanoid robot control systems are based on zero moment point and centre of gravity (COG) [16–18]. However, the balance control of an exoskeleton is executed by the pilot, and accurate COG parameters are not easily obtained. In addition, many learning-based online trajectory generation methods are proposed, which based on the physical human–robot interactions [19,20]. Thus, these online generation algorithms are not suitable for rehabilitation lower exoskele-

✉ Hong Cheng
  hcheng@uestc.edu.cn

[1] Center for Robotics, School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

tons, since the paraplegic patients can not provide active physical interactions with the lower exoskeleton. Therefore, reference trajectory generation in applications of rehabilitation requires kinematic information of the lower exoskeleton. Especially, most paraplegic patients can be trained to generate movement in their hip joints by using the inertia of their pelvises. However, obtaining the optimal gait trajectories of a lower extremity exoskeleton from hip joint movements has seldom been considered.

In this paper, we propose a novel technique for reference trajectory generation, which obtains the angle trajectories of knee joints from the hip joints. The angle relationships between the knee and hip joints are obtained from the kinematic models of the lower limb exoskeleton. The kinematic models require the position trajectories of the hip and ankle joints, whose parameters are optimised by an artificial fish swarming algorithm with a variable population. The proposed approach is validated on a simulated prototype and a physical prototype of the exoskeleton system named AIDER (AssistIve DEvice for paRalyzed patient).

The contributions of our research are threefold. First, we propose a novel reference trajectory generation algorithm that obtains the angle trajectories of knee joints from the hip joints. Second, we optimise the parameters of the reference trajectories by applying the artificial swarm fish algorithm with variable population (AFSAVP). Third, we evaluate the proposed reference trajectory generation algorithm on AIDER exoskeleton designed for this purpose.

The remainder of this paper is organized as follows. Section 2 presents the kinematic models of the lower limb exoskeleton. The position trajectory descriptors of the hip and ankle joints and the optimization algorithm are described in Sect. 3. Section 4 introduces the AIDER exoskeleton system. Experimental results are presented in Sect. 5. The paper concludes with Sect. 6.

2 Kinematic Models

The relationship between angle and position of the exoskeleton joints are obtained from the kinematic models of the exoskeleton in the sagittal plane. Since the inverse kinematic problem yields multiple solutions, constraints are added by automatically selecting segment points. Ultimately, the kinematic exoskeleton model derives the angular relationships between the hip and knee joints.

2.1 Forward Kinematic

Since the double support phase is very much shorter than the swing and stance walking phases, walking can be modelled as a two-phase cycle of alternating swings and stances. Furthermore, the balance of the exoskeleton is controlled by the
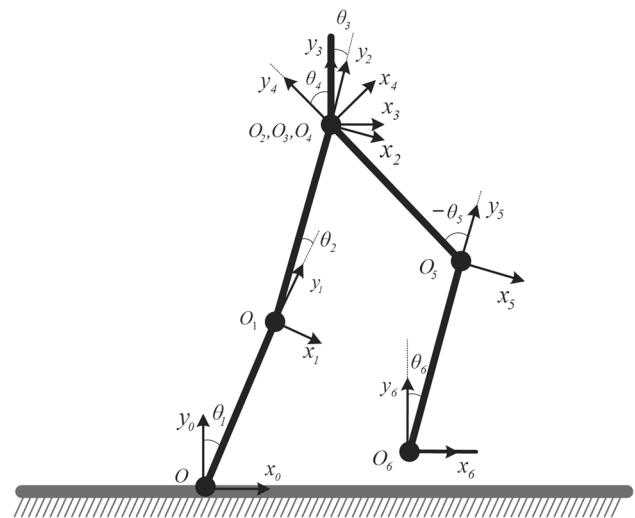
Fig. 1 The coordinates frames of the lower limb exoskeleton in the sagittal plane. Origins and coordinates axes of reference frames $1-6$ are denoted O_i and (x_i, y_i), respectively

wearer or peripheral equipment; thus, we consider motion in the sagittal plane alone. The reference frames and segment properties of the kinematic exoskeleton model in the sagittal plane are presented in Fig. 1. Two assumptions are made in this study, the trunk of the exoskeleton torso is vertical and the plantar of the exoskeleton is parallel with the ground during walking. Figure 1 also shows the coordinate systems of the lower limb exoskeleton. The inertial reference frame (Frame 0; origin denoted by O) is fixed on the ankle joint in the stance leg, which depends on the system state. Frame i and O_i (where $i = 1, 2, \ldots, 6$) denote the reference coordinate systems and their origins, respectively, fixed on the joints of the lower limb exoskeleton. The coordinate axes of frame i and the rotation angle between frames i and $i - 1$ are represented by $(x_i, y_i)(i = 1, 2, \ldots, 6)$ and $\theta_i(i = 1, 2, \ldots, 6)$, respectively.

The kinematic model of the exoskeleton shown in Fig. 1 is implemented by transformation matrices. Since motion occurs in the sagittal plane, the forward kinematic transformation matrices from frame $i - 1$ to i are given as follows:

$$
\begin{aligned}
T_i^{i-1} &= Rot(z; \theta_i) \cdot Trans(x, y; \alpha_i, \beta_i) \\
&= \begin{bmatrix} R_i^{i-1} & P_i^{i-1} \\ O_{1 \times 3} & 1 \end{bmatrix},
\end{aligned} \tag{1}
$$

where (α_i, β_i) represent the coordinate displacement of frame i relative to frame $i - 1$, and θ_i represents the rotation of frame i relative to frame $i - 1$ around the z coordinate. R_i^{i-1} is a 3×3 matrix of the orientations of frame i relative to frame $i - 1$, P_i^{i-1} expresses the 3×1 position vector of frame i relative to frame $i - 1$.

The transformation matrix between frame i and frame k $(k - i > 1)$ is given as follows:

$$T_k^i = T_{i+1}^i \cdot T_{i+2}^{i+1} \cdot \ldots \cdot T_{k-1}^k (i < k). \tag{2}$$

The relationships between each joint are determined from the position vectors of the hip and ankle joints in both stance and swing legs (P_3^0 and P_6^3, respectively). From Eqs.(1) and (2), we observe that the position vector $P_k^i (i < k)$ is part of the transformation matrix $T_k^i (i < k)$. Therefore, we obtain the relationships between hip and ankle joints by calculating T_3^0 and T_6^3. The position vectors P_3^0 and P_6^3 are calculated as follows:

$$\begin{cases} P_3^0 = \begin{bmatrix} -L_s \cdot \sin(\theta_1) - L_t \cdot \sin(\theta_1 + \theta_2) \\ L_s \cdot \cos(\theta_1) + L_t \cdot \cos(\theta_1 + \theta_2) \\ 0 \end{bmatrix} \\ P_6^3 = \begin{bmatrix} L_s \cdot \sin(\theta_4 + \theta_5) + L_t \cdot \sin(\theta_4) \\ -L_s \cdot \cos(\theta_4 + \theta_5) - L_t \cdot \cos(\theta_4) \\ 0 \end{bmatrix} \end{cases} \tag{3}$$

The angular trajectory of the hip joints is assumed available in the kinematic models, because trained paraplegic patients can generate hip movement from the inertia of their pelvises. In calculating the angular trajectory of the knee joint, we impose additional constraints by focusing on the y coordinate of the sagittal plane, enabling easier computation of the position trajectories' descriptors.

To more specifically describe the angel of each joint, we represent the angel of each joint in the stance and swing leg by $\theta_{aj}, \theta_{kj}, \theta_{hj} (j = 1, 2)$ instead of $\theta_i (i = 1, 2, \ldots, 6)$ in Eq. (3). The substitutions are presented in Table 1.

Therefore, the angular trajectory of the knee joint is obtained from the following relationship between the trajectories of the hip and ankle positions; it is given as follows:

$$L_s \cdot \begin{bmatrix} \cos(\theta_{a1}) \\ \cos(\theta_{h2} + \theta_{k2}) \end{bmatrix} + L_t \cdot \begin{bmatrix} \cos(\theta_{a1} + \theta_{k1}) \\ \cos(\theta_{h2}) \end{bmatrix}$$
$$= (Y_h - Y_a) \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \tag{4}$$

Table 1 The substitution of $\theta_i (i = 1, 2, \ldots, 6)$ in angular determination of stance and swing legs

$\theta_i (i = 1, 2, \ldots, 6)$	$\theta_{aj}, \theta_{kj}, \theta_{hj} (j = 1, 2)$
θ_1	θ_{a1}
θ_2	θ_{k1}
θ_3	θ_{h1}
θ_4	θ_{h2}
θ_5	θ_{k2}
θ_6	θ_{a2}

where Y_h and Y_a represent the vertical distance from the hip joint and the ankle joint to the ground, respectively.

From the assumption of vertical trunk and ground-parallel plantar during walking, we impose an extra constraint on the joints angle:

$$\theta_{hj} - \theta_{kj} - \theta_{aj} = 0, \qquad (j = 1, 2) \tag{5}$$

We can now compute the relationship between the angular trajectories of the hip and knee joints. Given constraint (5), Eq. (4) can be rewritten as the following single equation:

$$L_s \cdot \cos(\theta_h + \theta_k) + L_t \cdot \cos(\theta_h) = Y_h - Y_a, \tag{6}$$

where θ_h and θ_k denote the angles of the hip and knee joints, respectively, throughout the walking cycle. Note that Eq. (6) relates the hip and knee joint angles for both the swing leg and the stance leg.
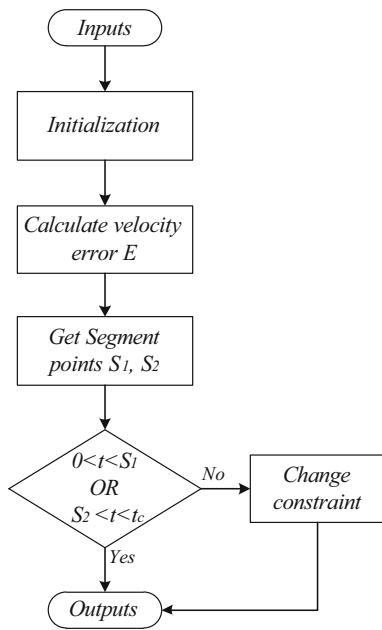
2.2 Eliminating Singular Solutions of the Inverse Kinematic

Because we focus only on the vertical coordinate of the sagittal plane, the inverse kinematic problem formulated in Eq. (6) leads to multiple solutions. To obtain a suitable solution to this nonlinear equation, we impose further appropriate constraints. A geometric description of the multiple solution problem (which admits Cases 1 and 2) is presented in Fig. 2. Cases 1 and 2 are determined from the constraints, which switch when the shank of the swing leg becomes perpendicular to the ground.

Singular solutions of the inverse kinematic equation are eliminated by an off-line training method. A flowchart of the off-line training is shown in Fig. 3. Input data are the whole gait cycle data, comprising the angular trajectories of the
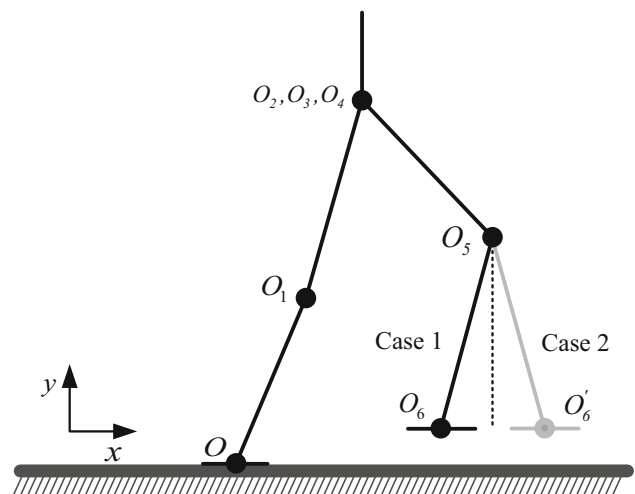
Fig. 2 Geometric description of the multiple solution problem

Fig. 3 Off-line training process for eliminating the singular solutions of the inverse kinematic

hip joints and the position trajectories of the hip and ankle joints. Singular solutions are removed by the additional two constraints:

$$\begin{cases} \theta_h + \theta_k \in [0, \pi], & 0 < t < S_1 \,\&\&\, S_2 < t < t_c \\ \theta_h + \theta_k \in [\pi, 2\pi], & S_1 < t < S_2 \end{cases} \quad (7)$$

where S_1, S_2 are set as two segment points and t_c represents the time of a whole gait cycle.

The initialization step calculates the angle trajectories of the knee joints by Eq. (6), imposing the first constraint in Eq. (7). The segment points are automatically chosen from the differential between the initial and generated angular velocities of the knee joints (velocity error E). The off-line training process outputs the angle trajectories of the knee joints outside the segment points ($0 < t < S_1$ and $S_2 < t < t_c$) and applies the second constraint in Eq. (7) to obtain the angle trajectories of knee joints inside the segment points ($S_1 < t < S_2$). Finally, the angle trajectories of the hip and knee joints are correctly related throughout the gait cycle.

The relationships between the angle trajectories of the hip and knee joints are evaluated in the online experiments, as discussed in Sect. 5.

3 Trajectory Generation and Parameter Determination

Recall that the hip and ankle joint positions are assumed to be known in the kinematic models. Therefore, to solve the

inverse kinematic problem described in Sect. 2.2, we first compute the position trajectories of the hip and ankle joints by trajectory generators. Next, the best description of these trajectories is selected by the AFSAVP.

3.1 Trajectory Description

In solving Eq. (6), the available parameters are the hip and ankle position trajectories along the vertical coordinate of the sagittal plane. Because the hip and ankle positions are difficult to measure, their trajectories are generated by two generators. These trajectory generators are suitable for wearers of different heights.

To simplify the problem, the ankle position trajectory generators are based on sinusoidal functions. The ankle motion generator proposed by Silva and Machado gives the following trajectory of the ankle position [21]:

$$\begin{cases} Y_a = \dfrac{H_a}{2}[1 - cos(2\pi \cdot f \cdot t)] + L_a, & \text{Swing Phase} \\ Y_a = L_a, & \text{Stance Phase} \end{cases} \quad (8)$$

where f is the step frequency and H_a and L_a represent the highest and the lowest position of the ankle during walking, respectively.
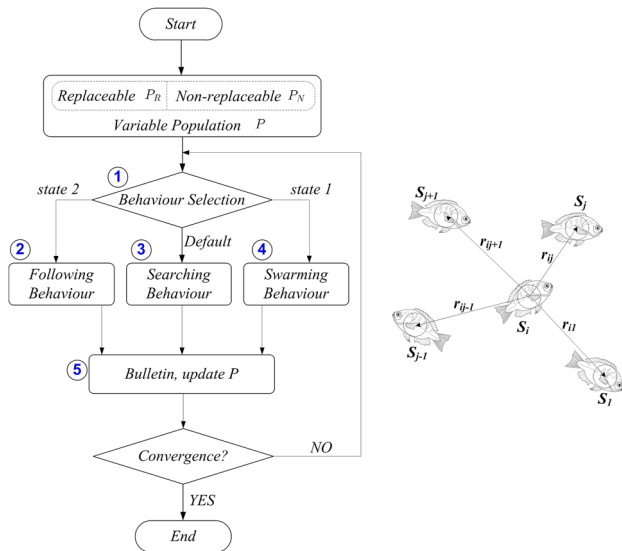
While walking, the vertical hip trajectory is treated as an inverted pendulum [22]. Yoon *et al.* approximated the path of the hip by three continuous differentiable polynomial segments [23]. The trajectory generator proposed in this paper is based on three sinusoidal functions, representing three components of the walking cycle. These functions are described as follows:

$$\begin{cases} Y_h = H_h + a_1 \cdot \sin\left(\pi \cdot t \cdot \dfrac{t_c}{t_{s1}}\right) - b_1, & 0 < t < t_{s1} \\ Y_h = H_h + a_2 \cdot \sin\left(\pi \cdot t \cdot \dfrac{t_c}{t_{s2}-t_{s1}}\right) - b_2, & t_{s1} < t < t_{s2} \\ Y_h = H_h + a_3 \cdot \sin\left(\pi \cdot t \cdot \dfrac{t_c}{t_c-t_{s2}}\right) - b_3, & t_{s2} < t < t_c \end{cases}$$
$$(9)$$

where H_h represents the distance from the hip to the ground when standing, t_{s1} and t_{s2} are separate points of the walking cycle, a_i and b_i ($i = 1, 2, 3$) are the adaptive parameters. In Eq. (8) and Eq. (9), H_a, L_a and H_h are related to height of the wearer [24].

3.2 Trajectory Parameter Determination using a Fish Swarming Algorithm

In order to achieve optimal parameters in Eq. (8) and (9), optimization methods should be employed. Genetic Algorithm (GA) is powerful in solving the optimization problems, which has already been applied in humanoid robots for biped loco-

Fig. 4 The AFSAVP flow chart and states of AF individuals

Table 2 Parameters of fitness function

Trajectory	Parameter(s)
Hip position	$a_i, b_i \ (i = 1, 2, 3)$
Ankle position	f

Table 3 Parameters of the SwarmFish optimization

Max generations	50
Test number	50
Experiment number	50
Visual factor	2.5
Crowd factor	0.618
Population	30
Try number	5
α	$[-1,1]$
β	$[-1,1]$
γ	$[300,500]$
ζ	$[1,6]$

motion [18]. However, the Genetic Algorithm would easily converge into the locally optimal solutions since the models are separated in various parts along timeline. Moreover, GA also get less efficiency in our cases, some simulation results will be discussed later. In this study, artificial swarm fish algorithm with variable population (ASFAVP) is employed to obtain global optimums and convergence faster.

Inspired by the collective intelligence of fish schooling behaviour, researchers have developed the basic artificial fish swarm algorithm (AFSA) an artificial intelligence algorithm that simulates the behaviour of an individual artificial fish (AF) and then constructs an AF school. We adopt the variable population AFSA (the above-mentioned AFSAVP) to generate the optimal parameters for the trajectory determination [25–27]. The AFSAVP converges more rapidly than the basic AFSA and is suitable for optimising the trajectory of the developed AIDER exoskeleton system.

The AFSAVP, which simulates optimal food searching under given initial conditions, is shown as a flowchart in Fig. 4. The five steps of the AFSAVP are (1) behaviour selection, (2) following behaviour, (3) searching behaviour, (4) swarming behaviour and (5) bulletin. The AFSAVP terminates under the condition 'max-generation'a commonly used criterion for terminating evolutionary processes.

As shown in the flowchart, the variable population P of the AF school comprises the replaceable school P_R and the non-replaceable school P_N, i.e. $P(t) = P_R(t) + P_N(t)$. This division of the population enhances the computational efficiency of the algorithm. The right part of Fig. 4 illustrates the behaviours of the fish school. During Step 1,these behaviours are selected by the food density; in Step 2, an AF follows his peers who have found the area of highest food density; in Step 3, he randomly moves from S_i to S_j and updates

his environmental conditions; in Step 4, he remains close to his neighbours within his visual field; in Step 5, the bulletin operation compares the current state S_i of each AF with the historical states and replaces the current state with a better state. To minimize the error in the generated trajectories, we set the fitness function F of the optimal trajectory to the following negative error function:

$$F(a_i, b_i, f) = -\sum_{i=1}^{3} |G_t(a_i, b_i, f) - O_t|, \tag{10}$$

where G_t represent the generated trajectories with parameters $a_i, b_i, f (i = 1, 2, 3)$, and O_t represents the original trajectories. The parameters of the fitness function to be optimised by AFSAVP are listed in Table 2. The AFSAVP-driven hybrid modelling of the trajectory parameter behaviours are simulated by the *SwarmFish* [27] toolbox. The parameters for the trajectory prediction are listed in Table 3. *Maxgenerations* specifies the termination condition of each test, and *experimentnumber* is the number of tests performed. Other parameters are the visual and crowd factors; the population size (comprising the numbers of non-replaceable P_N and replaceable population P_R individuals) and the trial number. Figure 5 shows the time course of the optimisation process of the fitness function. The 'max', 'mean' and 'min' fitness curves plateau during Generations 1–8. Note that all lines converge in generation 10, demonstrating the efficiency and accuracy of the proposed optimization algorithm.

In order to evaluate the efficiency of the proposed AFSAVP, we compared the proposed algorithm with basic
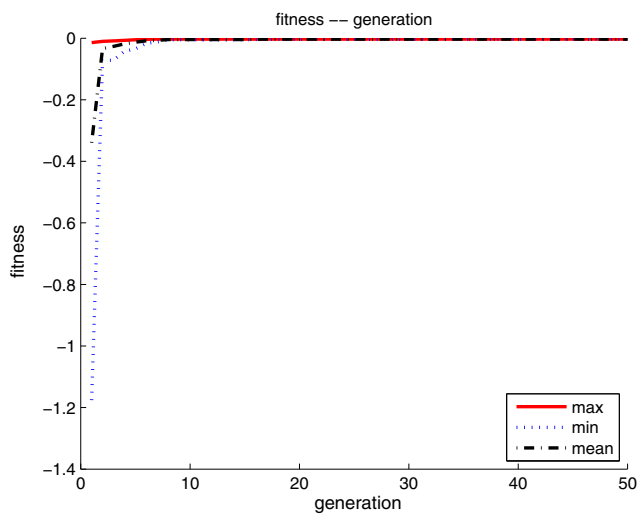
Fig. 5 Temporal progress of the optimisation of the fitness function by AFSAVP

Table 4 Comparison of AFSAVP, basic AFSA and Genetic Algorithm (GA)

| Generations	AFSAVP	AFSA	GA
Hip trajectory	9	16	24
Ankle trajectory	10	16	28

AFSA and Genetic Algorithm (GA) [18]. Comparison of optimisation efficiency of these algorithms are demonstrated in Table 4. In the simulation experiments, the original ankle and hip trajectories are utilized as reference trajectories. The experimental results indicate that the proposed AFSAVP can obtain the optimal parameters faster than other algorithms.

Figure 6 compares the AFSAVP-generated and original (experimental) ankle and hip trajectories. The AFSAVP generated trajectories remain close to the real trajectories, and the error range is acceptable for practical use. As showed in Fig. 6a, there is a light difference between the original (experimental) and AFSAVP-generated ankle position trajectories, since the model of AFSAVP-generated the ankle position trajectory. As discussed in Sect. 5, this difference exerts little effect on the generated angular trajectories of the knee joints.

4 The AIDER Exoskeleton System

4.1 Mechanical Structure of the AIDER Exoskeleton

To assist human walking, we have constructed an exoskeleton system called 'AIDER' (AssItive DEvice for paRalyzed patient). The AIDER exoskeleton is presented in Fig. 7. The rigid spine of the exoskeleton provides a tight connection to the wearer's upper body ('A' in Fig. 7). The three segments
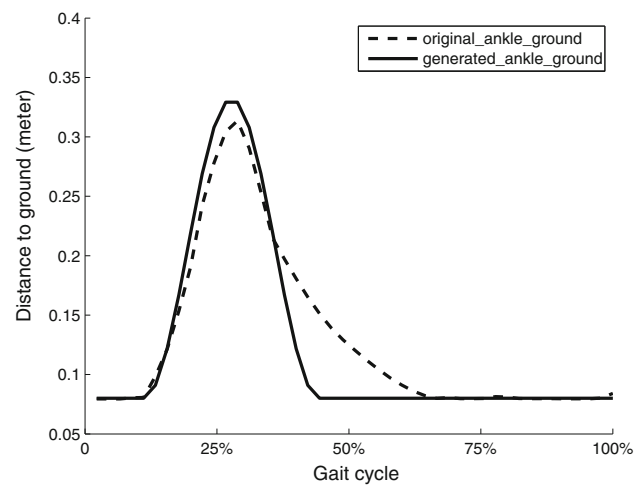
(a) ankle trajectories
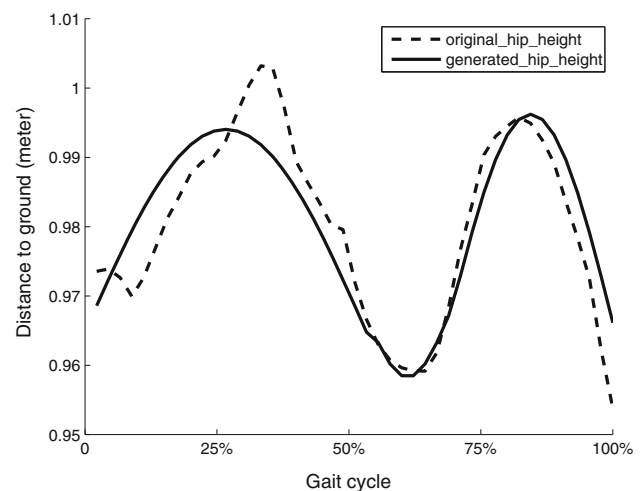
(b) hip trajectories

Fig. 6 Comparison of generated and original position trajectories **a** ankle trajectories **b** hip trajectories

are analogous to the wearer's lower limbs, and they retrieve analogous joint positions when the exoskeleton is worn by the wearer. The AIDER exoskeleton has ten single degrees-of-freedom (DOF) revolute joints, allowing five DOFs for each leg. An active torque at the hip?Cknee flexion of each exoskeleton leg ('D', 'E' in Fig. 7) is provided by a harmonic driver system comprising a DC servo motor and harmonic diver reduction. Additional passive DOFs are set at the hip abduction, ankle flexion and ankle abduction points ('C', 'F', 'G' in Fig. 7).

Besides establishing a tight connection with the wearer's torso, the exoskeleton forms flexible connections at the thighs, shanks and feet ('B' in Fig. 7). The torso is connected by a type of vest that distributes the forces between the AIDER exoskeleton and the wearer. The vest is made of rigid structures that support the wearer's upper body, and thick fabric that compliantly connects to the wearer.

Fig. 7 Mechanical structure of the AIDER exoskeleton, showing the degrees of freedom

A. **Compliant vest between PRMI exoskeleton and wearer**

B. **Compliant connections on thighs and shanks**

C. **Hip Abduction/Adduction (Passive)**

D. **Hip Flexion/Extension (Active)**

E. **Knee Flexion/Extension (Active)**

F. **Ankle Abduction/Adduction (Passive)**

G. **Ankle Plantarflexion/Dorsiflexion (Passive)**

Since the exoskeleton is a wearable robot, its kinematics should mimic human kinematics, and the motional ranges of each DOF should be approximated from physiological data [24]. Slight kinetic differences between the exoskeleton and human are acceptable because of the flexible connections between the AIDER exoskeleton and its wearer.

To provide effective physical support and walking assistance, the AIDER exoskeleton is installed with a real-time control system. The physical support provided by the control system depends on the weakness of the wearer's lower limbs, which is evaluated from the motion information detected by various sensors; namely, the encoders, posture measuring instruments and plantar sensors. For the wearers whose lower limbs are completely disabled, the control system provides the desired functional motion from the detected motional information.

4.2 Control Scheme of the AIDER Exoskeleton

Figure 8 shows the overall architecture of the AIDER exoskeleton control system hosting the control algorithm. This control system is divided into various networks, enabling effective integration of the information sent from the distributed sensors, thus reducing the complexity of the wiring. Consequently, the whole system achieves a high-speed, real-time control.

As shown in Fig. 8, the whole control system is divided into four sub-systems; a back control network, a control network for each leg and a graphic user interface (GUI) on a personal computer (PC). The principal network links are the connections between the control networks of the back and legs. Each link comprises two node controllers. The GUI can monitor the AIDER exoskeleton and replace the main con-

troller as the control network when the wearer requires more specific operations, which can be programmed on the PC.

## 5 Experimental Results and Discussion

5.1 Simulations

The availability of the generated angle trajectories is validated using the co-simulation platform written in ADAMS and MATLAB/Simulink. The ADAMS and MATLAB/Simulink software enable a versatile co-simulation platform, which not only accurately implements the mechanical system of the lower limb exoskeleton but also handles the trajectory generation algorithm in real-time [28]. A flowchart of the trajectory generation algorithm is presented in Fig. 9.

As depicted in Fig. 9, the relationship between the angle trajectories of hip and knee joints is determined in the off-line training module. In the simulation experiments, input data were obtained from a 180 cm-tall pilot walking through 10 gait cycles. Input data to Simulink are the kinematic parameters of the pilot and the angle trajectories of the hip joints throughout whole gait cycles. Figure 10 plots the average hip angle trajectory over the ten gait cycles, and the kinematic parameters are listed in Table 5. The thigh and shank lengths are given by $0.245 * H$ [24], where H is the height of the human body.

Afterwards, the position trajectories of the hip and ankle joints presented in Sect. 3.2 are obtained by the AFSAVP. From the optimised trajectories, we derive the relationship between the angle trajectories of the hip and knee joints. Finally, we relate these trajectories by the reference trajectory generation algorithm proposed in Sect. 2. The results are presented in 11. Figure 11a presents the error range of

Fig. 8 Overall architecture of the control system of the AIDER exoskeleton
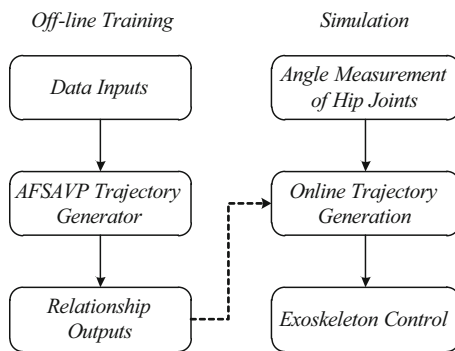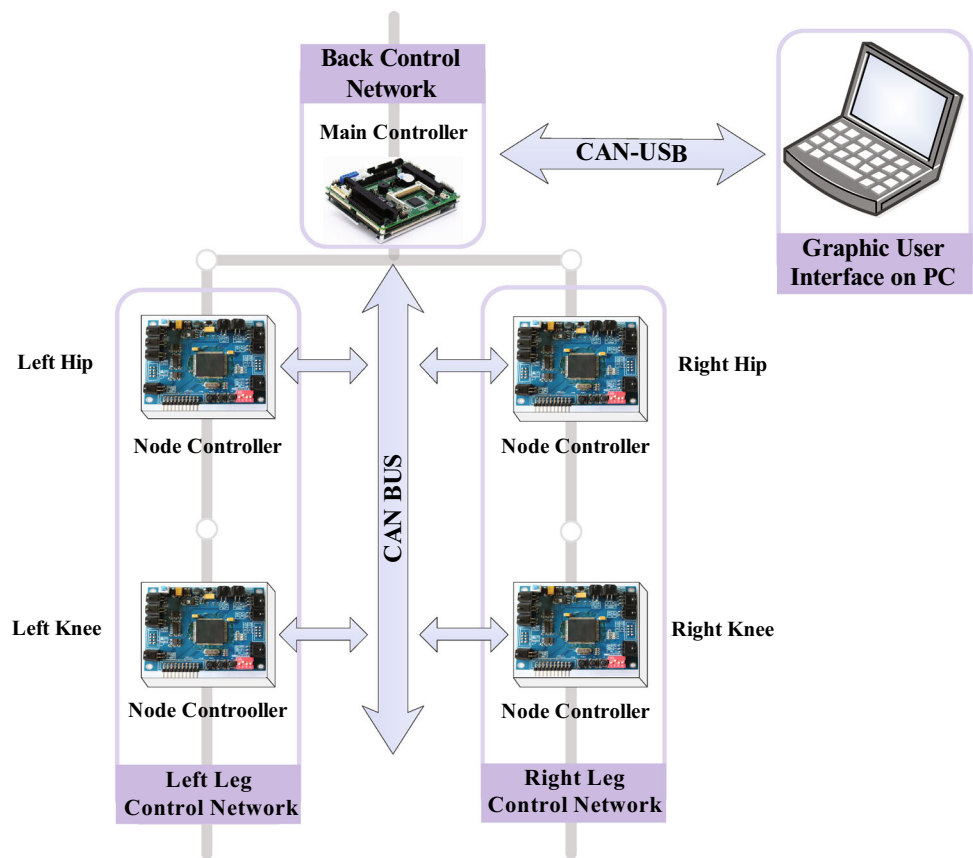
Fig. 9 The simulation procedure of trajectory generation algorithm in ADAMS and MATLAB/Simulink

The average of input hip angle trajectory in ten gait cycles
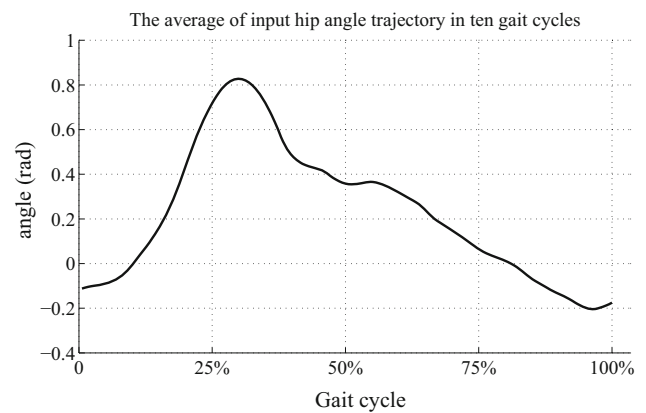
Fig. 10 Average hip angle trajectory throughout one gait cycle ($n=10$)

the generated hip-knee angle relationships throughout the ten gait cycles. The error range is large because different step lengths taken by the pilot affect the motional limits at each joint. The average generated relationships are compared with the original relationships (obtained from the 180-cm tall pilot) in Fig. 11b. The difference between the two plots is attributed to the model assumptions, which restrains the plantar of the pilot to remain parallel with the ground.

The online simulation process runs on the co-simulation platform implemented by ADAMS and Simulink. The relationship between the angle trajectories of hip and knee joints is applied to the online trajectory generation along with the

Table 5 Kinematic parameters for exoskeleton model in simulation environment

Parameters	Value (centimeters)
Height of pilot (H)	180
Length of thigh (L_t)	44
Length of shank (L_s)	44

hip joint angles that are measured by the virtual sensors in ADAMS. The generated angle trajectories of the knee joints are shown in Fig. 12.

The relationship between angle of hip joint and knee joint

(a)

The relationship between angle of hip joint and knee joint

(b)
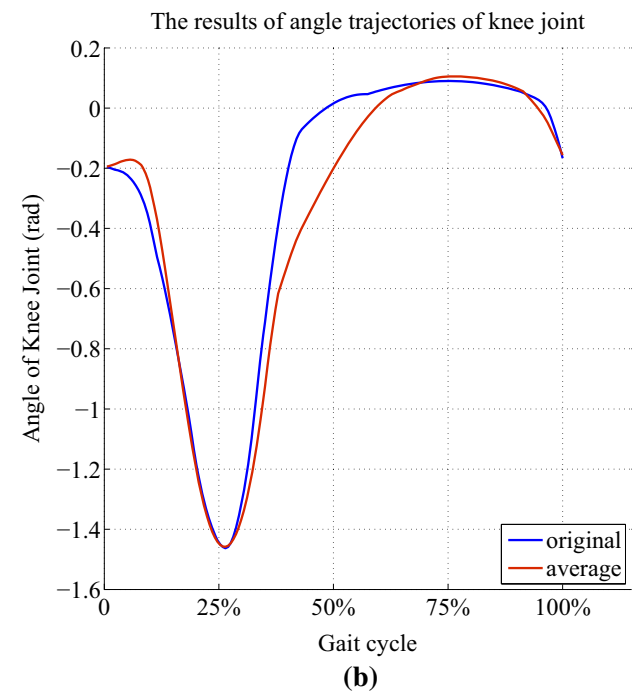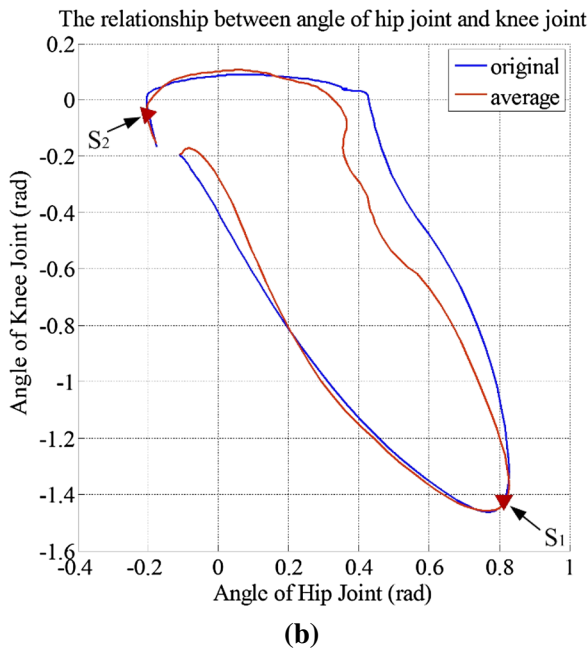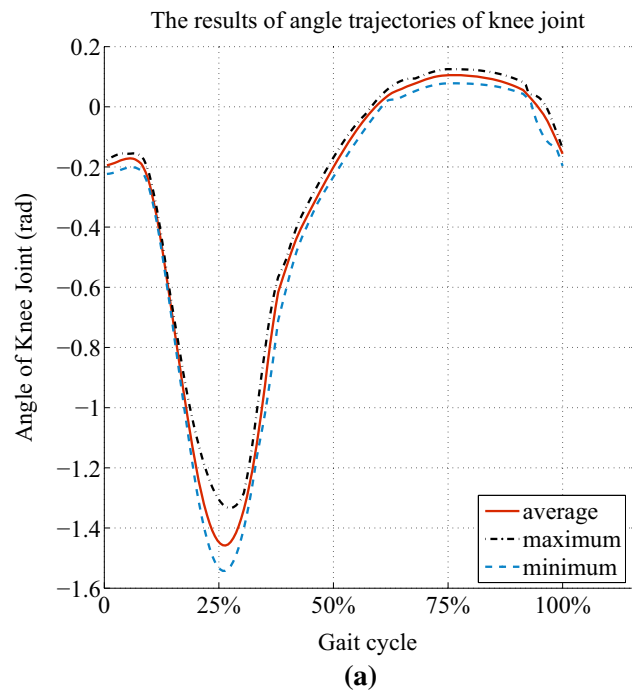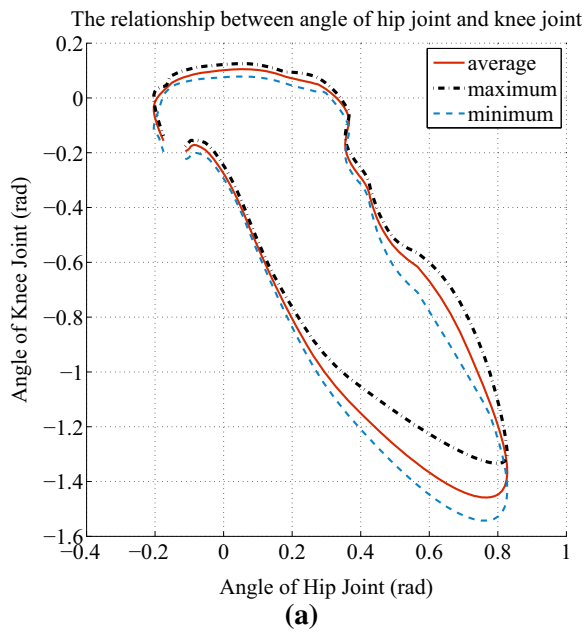
Fig. 11 The relations between the angles of the hip and knee joints: **a** Error range of the generated relationships throughout ten gait cycles; **b** Comparison between average and original hip-knee angle relationships. S_1 and S_2 indicate the segment points used to eliminate singular solutions of the inverse dynamics

The results of angle trajectories of knee joint

(a)

The results of angle trajectories of knee joint

(b)

Fig. 12 Generated angle trajectories of knee joints: **a** Error range of the generated knee angle trajectories throughout ten gait cycles; **b** Comparison of the average generated and original knee angle trajectories

The error range of the knee angle trajectories generated throughout ten gait cycles is shown in Fig. 12a. The differences in the swing phase result from the offline-determined relationship between the hip and knee angle trajectories. Figure 12b compares the average generated and original knee angle trajectories. Analysing the results, we find that when the heel strikes during the stance phase of normal walking,

the ankle joint of the exoskeleton undertakes a redundant movement that contradicts our assumption (that the plantar of the pilot is parallel with the ground). The error in the simulated trajectory generation throughout ten gait cycles is summarised in Table 6. The average error rate, 3.5 %, is within acceptable limits.

Table 6 Error analysis of the trajectory generation algorithm throughout ten gait cycles

Max error	0.33 (rad)
Average error	0.08 (rad)
Average error rate	3.5%
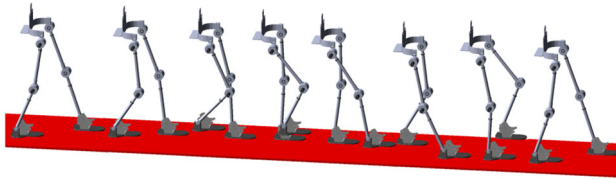
(a)

(b)

Fig. 13 Experimental results: **a** simulation and **b** real-time system. **a** The animation results in ADAMS software **b** The snapshots of whole gait cycle in real-time experiment on AIDER exoskeleton

To visualise the experimental results, we perform an animation in ADAMS software. The whole gait simulation shown in Fig. 13a demonstrates good walking performance of the exoskeleton. The error shown in Fig. 12b causes a vertical drop of the exoskeleton model during the early double stance phase in the animation results, but exerts little effect on the walking gait throughout the whole gait cycle.

5.2 Experiments on the AIDER Exoskeleton

5.2.1 Experimental Setup

To validate the algorithm on the real-time system, the AIDER exoskeleton was trialled on a normal person of height 180 cm. The pilot maintains balance but takes no active movement throughout the experiment.
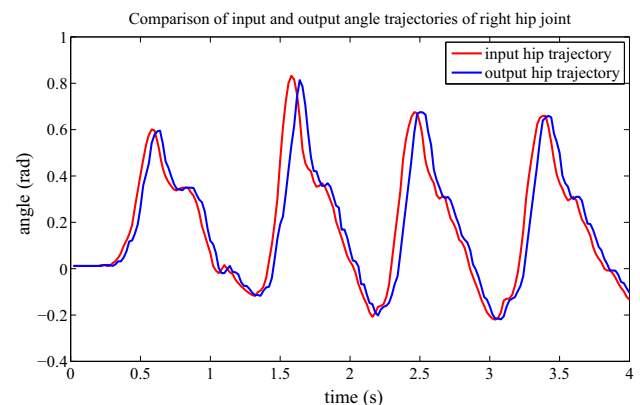
The angle trajectories of the knee joints are derived from the relationship between the angle trajectories of the hip and knee joints, which are trained off-line; specifically, the average relationships plotted in Fig. 11. During the AIDER exoskeleton trial, the hip joint angles were detected by angle sensors (which measure the joint angle data) in real-time. Based on the angle relationships between the hip and knee joints, the main controller runs the online trajectory generation algorithm. The knee-joint node controllers retrieve the knee angle trajectories from the CAN bus to control the motors at the knee joints. At each node controller, the movement accuracy is increased by a position control strategy with a PID controller.
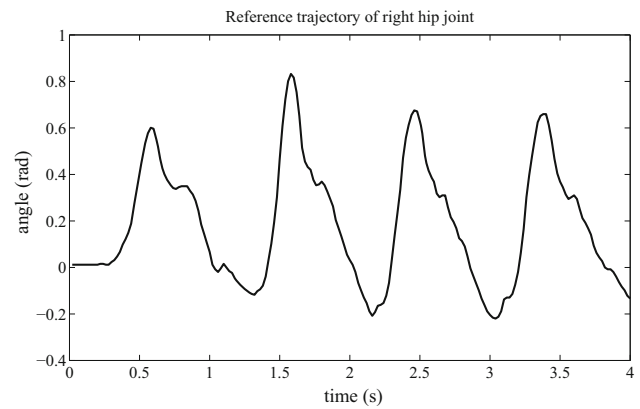
Fig. 14 Reference trajectory of right hip joint during normal walking

The experiment was performed through four gait cycles of normal walking, and results were collected from the right leg of the pilot and exoskeleton. The angle trajectory of the right hip joint, measured by angle sensors affixed to the pilot during normal walking, is shown in Fig. 14.

5.2.2 Results and Discussion

Figue 13b presents snapshots of the whole gait cycle during the AIDER exoskeleton trial. The motion of the right knee joint shows good performance during normal walking. As shown in Fig. 15, the output angle of the right hip joint, measured by the joint encoder on the AIDER exoskeleton, was slightly delayed relative to the input reference measured by angle sensors on the pilot. This delay is intrinsic, because there is a 50 ms lag in the control system of the AIDER exoskeleton. Figure 16 compares the input and output angle trajectories of the right knee joint, where the input trajectory is calculated by the trajectory generation algorithm running in the main controllers of the AIDER exoskeleton, and the

Fig. 15 Comparison of input and output angle trajectories of the right hip joint in a real-time experiment on the AIDER exoskeleton
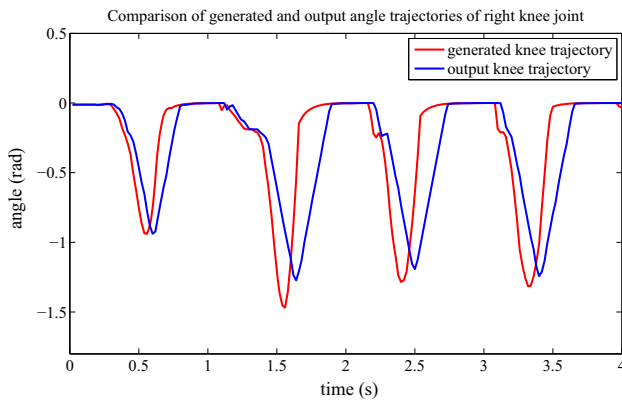
Fig. 16 Comparison of the generated and output angle trajectories of the right knee joint in a real-time experiment on AIDER exoskeleton

Table 7 Experimental results on the AIDER exoskeleton

Joint Name	Control lag (ms)	Max error (rad)	Average error (rad)	Average error rate
Right hip	50	0.28	0.07	1.74 %
Right knee	75	0.64	0.2	4.52 %

Table 8 Performances of node controllers and trajectories generators in the AIDER exoskeleton

Joint name	Tracking errors in node controller nMSE (rad)	Error with original trajectories nMSE (rad)
Right hip	0.04	/
Right knee	0.06	0.12

output trajectory is measured by the joint encoder affixed to the right knee.

As evidenced by the experimental results, the knee joint angle trajectory computed by the trajectory generation algorithm closely matches that of normal human walking. The performance of the AIDER exoskeleton during the experiment is summarised in Table 7. The error in the right knee joint is larger than in the simulation platform because the control lag in the real-time system introduces a time delay to the gait cycles. Although the control lag and trajectory generation algorithm collectively delay the output angle trajectory of the right knee joint by 75 ms, the whole system ensures real-time control throughout the experiment. Tracking errors of the node controllers which presented in Table 8 also indicate that the controller can achieve good performance with given trajectories. Moreover, the errors between the original and generated trajectories are also provided in Table 8, the results show that the generated knee trajectories are acceptable for controllers. According to the above discussions, the experimental results show that the proposed trajectory generation algorithm has a good performance on the AIDER exoskeleton.

6 Conclusions and Future Work

This paper introduces a novel reference trajectory generation algorithm based on the kinematics and inverse kinematics of human walking. The proposed algorithm obtains the angle trajectory of the knee joints from the angle trajectory of the hip joints. The optimal parameters for determining the hip and ankle position trajectories are computed by AFSAVP. The algorithms are validated on an ADAMS and MATLAB/Simulink co-simulation platform, which integrates algorithms and accurate mechanical systems. According to the simulation results, the average error in the knee angle trajectory is 4.5 %, and high performance was confirmed in an animation simulation. In the AIDER exoskeleton experiment, the joint angle trajectories generated by the reference trajectory generation algorithm accurately replicated those of normal human walking.

This paper investigated three main themes. First, the relationship between the angle trajectories of the hip and knee joints was captured in a kinematic model of the lower limb exoskeleton. Second, AFSAVP was proposed to obtain the optimal parameters of the position trajectories of the hip and ankle joints. Third, the reference trajectory generation algorithm was validated on a custom-built AIDER exoskeleton.

In future versions of the algorithm, the optimal angle trajectory of the hip and ankle should be acquired by online parameter learning. Currently, the parameters that optimise this trajectory are learned offline. Online learning will enable adaptation to wearers of different sizes and application to real-time systems. Future work will also implement and evaluate the proposed algorithm in rehabilitation programmes for paraplegic patients who can be trained to generate hip movements from the inertia of their pelvises.

References

1. Sankai Y (2011) HAL: hybrid assistive limb based on cybernics. In: International symposium on robotics reasearch, pp 25–34
2. Sakurai T, Sankai Y (2009) Development of motion instruction system with interactive robot suit HAL. In: IEEE international conference on robotics and biomimetics, pp 1141–1147
3. Bogue R (2009) Exoskeletons and robotic prosthetics: a review of recent developments. Ind Robot 36(5):421–427
4. Strausser KA, Kazerooni H (2011) The development and testing of a human machine interface for a mobile medical exoskeleton. In: IEEE/RSJ international conference on robots and systems, pp 4911–4916

5. Dollar AM, Herr H (2008) Lower extremity exoskeletons and active orthoses: challenges and state-of-the-art. IEEE Trans Robot 24(1):144–158

6. Kazerooni H, Steger R, Huang L (2006) Hybrid control of the Berkeley lower extremity exoskeleton (BLEEX). Int J Robot Res 25(5–6):561–573

7. Walsh CJ, Pasch K, Herr H (2006) An antonomous, under-actuated exoskeleton for load-carrying augmentation. In: IEEE international conference on robotics and automation, pp 1410–1415

8. Kazerooni H, Chu A, Steger R (2007) That which does not stabilize, will only make us stronger. Int J Robot Res 26(1):75–89

9. Zoss A, Kazerooni H, Chu A (2005) On the mechanical design of the Berkeley lower extremity exoskeleton (BLEEX). In: IEEE/RSJ international conference on intelligent robots and systems pp 3465–3472

10. Li Z, Ge SS (2013) Adaptive robust controls of biped robots. IET Control Theory Appl 7(2):161–175

11. Li Z, Su CY (2013) Neural-adaptive control of single-master-multiple-slaves teleoperation for coordinated multiple mobile manipulators with time-varying communication delays and input uncertainties. IEEE Trans Neural Netw Learn Syst 24(9):1400–1413

12. Wang C, Li Y, Ge SS, Lee TH (2015) Optimal critic learning for Robot control in time-varying environments. IEEE Trans Neural Netw Learn Syst 26(10):2301–2310

13. He W, Chen Y, Yin Z (2015) Adaptive neural network control of an uncertain robot with full-state constraints. In: IEEE transactions on cybernetics (in press). doi:10.1109/TCYB.2015.2411285

14. He W, Ge SS, Li Y et al (2012) Impedance control of a rehabilitation robot for interactive training. In: Ge SS, Khatib O, Cabibihan J-J et al (eds) Social robotics. Springer, Heidelberg, pp 526–535

15. He W, Ge SS, Li Y et al (2014) Neural network control of a rehabilitation robot by state and output feedback. J Intell Robot Syst 80(1):15–31

16. Huang Q, Yokoi K, Kajita S, Kaneko K, Arai H, Koyachi N, Tanie K (2001) Planning walking patterns for a biped robot. IEEE Trans Robot Autom 17(3):280–289

17. Vermenulen J, Verrelst B, Vanderborght B, Lefeber D, Guillaume P (2006) Trajectory planning for the walking biped 'Lucy'. Int J Robot Res 25(9):867–887

18. Udai A (2008) Optimum hip trajectory generation of a biped robot during single support phase using genetic algorithm. In: International conference on emerging trends in engineering and technology, pp 739–774

19. Yang C, Burdet E (2011) A model of reference trajectory adaptation for interaction with objects of arbitary shape and impedance. In: IEEE/RSJ international conference on robots and systems, pp. 4121–4126

20. Yang C, Ganesh G, Haddadin S et al (2011) Human-like adaptation of force and impedance in stable and unstable interactions. IEEE Trans Robot 27(5):918–930

21. Silva F, Machado J (1997) Kinematic aspects of robotic biped locomotion systems. In: Proceedings of IEEE/RSJ international conference on intelligent robots and systems, pp. 266–272

22. Zijlstra W, Hof A (1997) Displacement of the pelvis during human walking: experimental data and model predictions. Gait Posture 6(3):249–262

23. Yoon J, Kumar R, Özer A (2011) An adaptive foot device for increased gait and postural stability in lower limb Orthoses and exoskeletons. Int J Control Autom Syst 9(3):515–524

24. Herman I (2008) Physics of the human body. Springer, New York

25. Chen Y, Zhang GF, Li YY, Ding Y, Zheng B, Miao Q (2013) Quantitative analysis of dynamic behaviours of rural areas at provincial level using public data of gross domestic product. Entropy 15(1):10–31

26. Chen Y, Miao Q, Zheng B, Wu SM, Pecht M (2013) Quantitative analysis of lithium-ion battery capacity prediction via adaptive bathtub-shaped function. Energies 6(6):3082–3096

27. Chen Y (2011) SwarmFish-The artificial fish swarm algorithm. Available via online http://www.mathworks.com/matlabcentral/fileexchange/32022

28. Rahmfeld R (2003) Modeling of hydraulic mechanical systems cosimulation between ADAMS and Matlab/Simulink. Techinische Universitaet Hamburg-Harburg, Hamburg

Rui Huang received the M.S. degree in Automation from University of Electronic Science and Technology of China in 2010. Now he is a third-year Ph.D. candidate in Center for Robotics in University of Electronic Science and Technology of China. His research interests inlude exoskeleton control, reinforcement learning, hierarchical learning, and physical human–robot interaction.

Hong Cheng is a full Professor in School of Automation, a vice director of Center for Robotics, UESTC. He received Ph.D. degree in Pattern Recognition and Intelligent Systems from Xi'an Jiaotong University in 2003. Now he is the founding director of Pattern Recognition and Machine Intelligence Lab, UESTC. Before this, he was a postdoctoral at Computer Science School, Carnegie Mellon University, USA from 2006 to 2009. His current research interests include computer vision and machine learning, social robotics, human computer interaction, multimedia signal processing. Dr. Cheng has over 90 academic publications including two books. He has been a senior member of IEEE, ACM, and Associate Editor of IEEE Computational Intelligence Magazine. He is the local chair of the 3rd China-USA Vision, Learning and Pattern Recognition summer school (VLPR 2011), financial chair of International Conference on Multimedia & Expo (ICME 2014), general chair of Vision and Learning SEminar (VALSE 2015), financial chair of 3rd IEEE China Summit & International Conference on Signal and Information Processing (ChinaSIP 2015). Moreover, Dr. Cheng will be the program chair of Chinese Conference on Pattern Recognition (CCPR) in 2016.

Yi Chen is a lecturer in Dynamics and Control at Glasgow Caledonian University. Dr Chen is specialised in computational intelligence aided design, cyber-physical systems, robotics, dynamics & control, multidisciplinary design and optimisation under uncertainties, reliability and risk analysis, automotive systems, space tether systems, computational finance and optical engineering, etc. He attracted research funding from Nature Science Foundation of China, China Scholarship Council, KTP projects, the Innovation Fund, the Fundamental Research Funds for Central Universities research and produced over 50 publications in high quality, international, peer-reviewed journals and conferences in the past few years. Dr. Chen also studies how people construct evaluations and preferences in social contexts.

Qiming Chen is a second-year Ph. D candidate in robot center of University of Electronic Science and Technology of China. His research interests include exoskeleton robot, imitation learning, perception of robot, human–robot interaction.

Xichuan Lin received his Ph.D. degree in Intelligent Mechanical Systems Engineering from Kagawa University in Japan in 2012. Now he is working in the Machine Intelligence Institute, in School of Automation, UESTC. His current research interests include intelligent control systems, robotic systems, exoskeleton robots and dynamics modeling of robotics. Dr. Lin now has 14 academic publications including three SCI international journals, 10 IEEE conference papers and one book

chapters. He was the conference committee of IEEE ICMA 2009–2011 and IEEE/ICME CME 2011. He was also the reviewer for international conferences (IEEE ICMA, IEEE/ICME CME, IJMA, etc).

Jing Qiu received her Ph.D. degree in Mechanical Engineering from Darmstadt University of Technology. Now she is a vise professor of center for robotics in university of electronic science and technology of china. Her current works on human–robots interactions of an exoskeleton for assisting walking. Her work in these areas focuses on human factors/Ergonomics, physical Human–Robot interaction, human modelling.