



# Real-time walking pattern generation for a lower limb exoskeleton, implemented on the Exoped robot

Jafar Kazemi, Sadjaad Ozgoli\*

Department of Electrical and Computer Engineering, Tarbiat Modares University, Tehran, Iran



## HIGHLIGHTS

- A real-time walking pattern generation method is provided.
- The trajectory planning is formulated as a feedback control problem.
- A solution is provided for an optimal control problem with changeable final states.
- The backward balance of the lower limb exoskeleton is considered in the walking pattern.

## ARTICLE INFO

### Article history:

Received 30 September 2018

Received in revised form 10 January 2019

Accepted 18 February 2019

Available online 2 March 2019

### Keywords:

Exoskeleton

Walking pattern

Optimal control

Center of mass

Exoped

## ABSTRACT

Lower extremity exoskeletons have been developed as a motion assistive technology in recent years. Walking pattern generation is a fundamental topic in the design of these robots. The usual approach with most exoskeletons is to use a pre-recorded pattern as a look-up table. There are some deficiencies with this method, including data storage limitation and poor regulation relating to the walking parameters. In addition, the walking parameters can be taken in hand very hard. Therefore modeling the human walking pattern is required. The few existing models provide piece by piece walking patterns, only generating at the beginning of each stride. In this paper, a real-time walking pattern generation method is provided which enables changing the parameters during the stride. For this purpose, two feedback controlled third order systems are proposed as real-time trajectory planners for generating the trajectories of the x and y components of each joint's position. The boundary conditions of the trajectories are determined to prevent backward balance loss by appropriate displacement of the center of mass. In addition, a cost function is intended for each trajectory planner in order to increase the smoothness of trajectories. Optimization technique is used to design the feedback controller for tracking the boundary conditions in such a way that the cost function is minimized. Finally, the proper joint angles are generated using inverse kinematics transformation. The performance of the proposed pattern generator is verified via real experiments on the Exoped robot.

© 2019 Elsevier B.V. All rights reserved.

## 1. Introduction

Increase in diseases and accidents related to human mobility, in addition to population aging, has caused much attention to be given to motion assistive technologies including wearable robots; particularly lower limb exoskeletons. Research on powered human exoskeleton devices dates back to the 1960s in the United States [1] and in the former Yugoslavia [2] for military and medical service purposes respectively. Since then, exoskeleton robots have been well-developed, particularly for medical purposes. Some of them have even hit the market [3].

One of the major challenges in designing these robots is walking pattern generation. There are various methods for planning

walking patterns, depending on the application and structure of the exoskeleton. These methods can be classified into three categories: model-based, sensitivity amplification, and predefined gait trajectory strategies [4]. The model-based strategies take stability into account in order to determine a walking pattern. The zero moment point (ZMP) is the most widely used stability criterion in gait planning [5,6]. This method relies on the accuracy of the human-exoskeleton model and requires various sensors. Sensitivity amplification strategies are applied to the exoskeletons in which user applies force to the robot. These include load carrying [7] and rehabilitation [8] robots. Finally, predefined gait trajectory strategies use a pre-recorded pattern of a healthy person as the reference trajectory. These are proper mostly for robots that a patient is the user.

Due to the difficulty of accurate ZMP measurement, model-based strategies are not cost-efficient. Furthermore, sensitivity

\* Corresponding author.

E-mail address: [ozgoli@modares.ac.ir](mailto:ozgoli@modares.ac.ir) (S. Ozgoli).

amplification methods are not suitable for the kind of assistance which paraplegic patients need. Therefore, predefined gait trajectory methods are the usual approach in most of medical exoskeletons such as ReWalk [9], eLEGS [10], and ATLAS [11], which is aimed at subjects who are losing their ability to move. There are some shortcomings with this method such as data storage limitation and pattern adjustment depending on different walking parameters and different individuals. Motivated by the above problems, a number of simple models have been developed to generate walking patterns alternated to pre-recorded gaits. The common procedure for these models is to determine the boundary conditions of the joints' trajectory at some specified via-points and to fit a mathematical curve (e.g. polynomial or sinusoidal) to them. In [12] and [13], the hip trajectory is approximated by polynomial and sinusoidal function segments respectively. In [14], two sinusoidal trajectories are proposed as the x and y components of ankle position. In [15] polynomial and sinusoidal functions are used for generating the joint positioning of a biped. Also, spline interpolation are used in [16] for joint motion planning of a wearable robot by satisfying determined via-point constraints.

In previous works, the trajectories are generated piece by piece as the segments between via-points and the endpoint boundary conditions are fixed within the segments. In other words, the walking patterns are determined at the via-point times and changing walking parameters during a stride is not possible. While, for real-time control of the robots, the walking parameters may be updated at any time in facing obstacles or difficult situations. Therefore, an online pattern generator is required to adjust the trajectories according to the changes in the parameters.

In this paper, we propose a real-time minimum jerk walking pattern generator for lower limb exoskeletons. In the proposed method, the walking parameters (including step length, maximum foot clearance, and stride time) can be changed during the stride without discontinuity in the second derivative of the trajectories. The idea of our method is to consider the trajectory as the output of a third order system. A feedback controller is then designed to impose the system's output to satisfy the boundary conditions of the trajectory. By introducing a minimum jerk cost function, the trajectory planning problem is formulated as an optimal control problem with changeable final states. A solution for this problem is proposed using the minimum principle of Pontryagin [17].

It should be noted that the balance and the safety of the user is the most important concern in design of the medical lower limb exoskeletons. Stability of the robot is guaranteed when using the exoskeleton as a walking training assistance between parallel bars. While, for the personal applications and outside of the laboratories where the wearer is supported by crutches or walkers, the balance is maintained only in the forward direction. Therefore, there is not a support tool in the backward direction and the backward balance should be considered in the gait. In order to ensure the dynamic stability of the robot, we set the boundary conditions of the hip position in such a way to prevent a backward balance loss.

The remainder of this paper is organized as follows: Section 2 introduces the Exoped robot as our implementation platform. In Section 3, the real-time trajectory planners are designed for each x and y component of the position of the joints. In Section 4, the boundary conditions of the trajectories are determined in such a way to avoid a backward balance loss. The simulations and experimental results are presented in Section 5. Finally, the paper is concluded in Section 6.



**Fig. 1.** The Exoped robot and pilot.

## 2. The Exoped robot

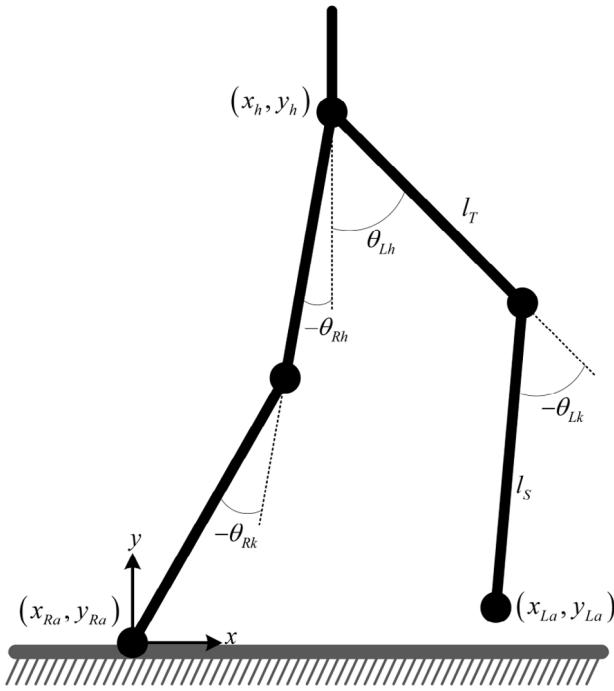
Exoped is a 4 DOF lower limb exoskeleton. It has 2 DOFs on each leg: 1 at the hip and 1 at the knee, driven by 4 brushless electronically communicated (EC) motors (Fig. 1). The ankle joints behave passively with a spring and their default angles are set at 90° relative to the leg. The motors from the Maxon company (model "EC 90 flat"), fed by 36 V power supply, are used. Each motor is coupled with a 1:135 gearbox and internal hall sensors are used to indicate the position. The control algorithm of the robot including state machine, walking pattern generation, and PID controller of the motors is realized in Simulink. The designed Simulink model is then converted to C code using the Wajjung library block set [18] and implemented on Stm32f429 microcontroller.

The forward kinematics of Exoped can be described as follows:

$$\begin{aligned}x_R &= l_T \sin(\theta_{Rh}) + l_S \sin(\theta_{Rh} + \theta_{Rk}) \\y_R &= -l_T \cos(\theta_{Rh}) - l_S \cos(\theta_{Rh} + \theta_{Rk}) \\x_L &= l_T \sin(\theta_{Lh}) + l_S \sin(\theta_{Lh} + \theta_{Lk}) \\y_L &= -l_T \cos(\theta_{Lh}) - l_S \cos(\theta_{Lh} + \theta_{Lk})\end{aligned}\quad (1)$$

where R and L refer to right and left leg, respectively.  $\theta_{*h}$  and  $\theta_{*k}$  denote the angles of the hip and knee joint of each leg.  $l_T$  and  $l_S$  represent the length of thigh and shin respectively and  $x_R, y_R, x_L$ , and  $y_L$  are expressed by:

$$x_R = x_{Ra} - x_h$$



**Fig. 2.** Robot parameters description.

$$\begin{aligned} y_R &= y_{Ra} - y_h \\ x_L &= x_{La} - x_h \\ y_L &= y_{La} - y_h \end{aligned} \quad (2)$$

in which  $\vec{r}_h = (x_h, y_h)$ ,  $\vec{r}_{Ra} = (x_{Ra}, y_{Ra})$ , and  $\vec{r}_{La} = (x_{La}, y_{La})$  represent the position of hip, right ankle and left ankle in sagittal plan, respectively. The initial position of the right ankle is defined as the origin point of coordination. The defined parameters are shown in Fig. 2.

The inverse kinematics equations of the robot are described as the following:

$$\begin{aligned} \theta_{Lk} &= \cos^{-1} \left( \frac{l_T^2 + l_S^2 - x_L^2 - y_L^2}{2l_T l_S} \right) - 180 \\ \theta_{Lh} &= \tan^{-1} \left( \frac{-x_L}{y_L} \right) + \cos^{-1} \left( \frac{x_L^2 + y_L^2 + l_T^2 - l_S^2}{2l_T \sqrt{x_L^2 + y_L^2}} \right) \\ \theta_{Rk} &= \cos^{-1} \left( \frac{l_T^2 + l_S^2 - x_R^2 - y_R^2}{2l_T l_S} \right) - 180 \\ \theta_{Rh} &= \tan^{-1} \left( \frac{-x_R}{y_R} \right) + \cos^{-1} \left( \frac{x_R^2 + y_R^2 + l_T^2 - l_S^2}{2l_T \sqrt{x_R^2 + y_R^2}} \right). \end{aligned} \quad (3)$$

### 3. Real-time trajectory planning

Fig. 3 shows the overall control schematic of the robot. As shown, a well-defined algorithm calculates the walking parameters according to the stability of the robot and particular conditions; e.g. patient's dimension and environment conditions. The walking parameters are used as the inputs for a pattern generator block. These parameters are step length, maximum foot clearance, and stride time denoted by  $L_s$ ,  $H_s$ , and  $T_s$ , respectively. The pattern generator provides the appropriate position of the joints in the sagittal plane and subsequently, the positions are transformed into joint angles using the inverse kinematics equations. Finally, a PID feedback controller is employed to regulate the joint angles.

The pattern generator consists of independent trajectory planners that generate the trajectory of the x and y components of each joint's position (Fig. 4). A boundary condition generator calculates the endpoint conditions of the trajectories in order to satisfy the walking constraints and maintain the balance of the robot. The vectors  $X_j(t)$  and  $Y_j(t)$  are defined as

$$\begin{aligned} X_j(t) &= [x_j(t) \quad \dot{x}_j(t) \quad \ddot{x}_j(t)]^T \\ Y_j(t) &= [y_j(t) \quad \dot{y}_j(t) \quad \ddot{y}_j(t)]^T \end{aligned} \quad (4)$$

where the symbol  $j$  indicates the joints and  $j \in \{h, Ra, La\}$ . As shown in Fig. 4 the boundary conditions are sufficient for planning the trajectory of  $x_j(t)$ . While, for generating the trajectory of  $y_j(t)$ , the peak value of the trajectory is also required to define the height of the joint in addition to the boundary conditions. The peak value is denoted by  $y_j^p$  which equals to  $H_s$  for the ankle joints and is determined for the hip joint according to the walking parameters.

Continuity, smoothness, and taking walking constraints into account are the main objectives to be considered in trajectory planning. In most previous works on pattern generation, the walking parameters are considered to be constant during the stride. The most common method of trajectory planning is fitting a mathematical curve (e.g. polynomial or sinusoidal) to some boundary conditions obtained by walking parameters. From a systems point of view, this kind of trajectory planner can be described as an input-output zero order system (Fig. 5). With this, discontinuity of the inputs caused by any change in walking parameters yields to discontinuity in the output.

By using the control scheme depicted in Fig. 3, the walking parameters may be updated at any time due to a change in the stability of the robot or the particular conditions. Therefore, the above-mentioned trajectory planner (as in Fig. 5) cannot be used and an online trajectory planner is required to adjust the trajectory according to the change in the parameters in order to maintain the second derivative continuity. For this purpose, we propose feedback controlled third order systems as the online trajectory planners for generating the trajectory of the joints' x and y component, as follows.

#### 3.1. The x component of the joints

Fig. 6 Shows a general trajectory shape for the x component of the position of the joints, starting from the initial condition  $X_j(t_0)$  converging to the final value  $X_j(t_f)$ , with the minimum curvature.

In order to generate this type of trajectory, we propose the feedback controlled third order system depicted in Fig. 7, which can be formulated as follows:

$$S_x: \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 \\ \dot{x}_3 = u \\ x_j = x_1 \end{cases} \quad (5)$$

It is obvious that a finite  $u$  yields a continuous trajectory with a continuous second derivative. In addition, the cost function denoted by  $J_x$  is intended to be minimized in order to increase the smoothness of the trajectory.

$$J_x = \int_{t_0}^{t_f} \left( \frac{d^3 x_j}{dt^3} \right)^2 dt \quad (6)$$

As a result, the real-time trajectory planner can be designed by calculating the feedback control law  $u$  to move the states of the system  $S_x$  to the changeable final condition  $X_j(t_f)$  in such a way that the cost function  $J_x$  is minimized. By "changeable" we mean it can be changed during the time. We have presented Theorem 1 to solve this optimal control problem. Lemma 1 is used to prove Theorem 1.

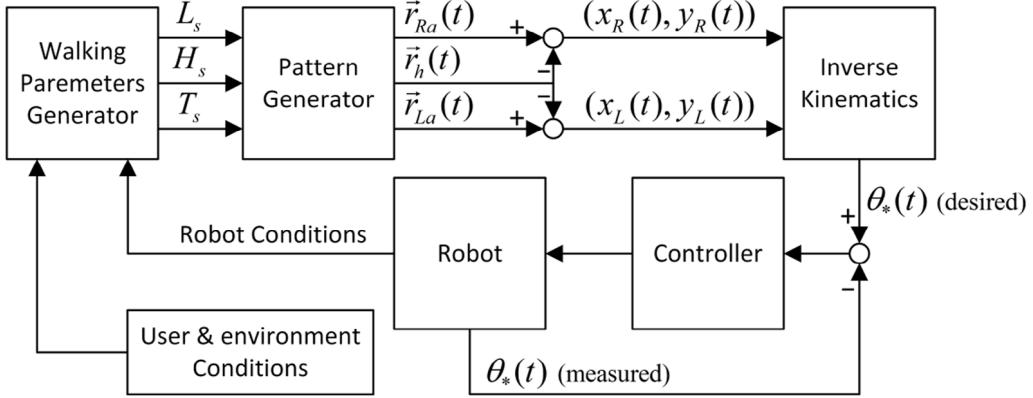


Fig. 3. The overall control schematic of the exoskeleton robot.

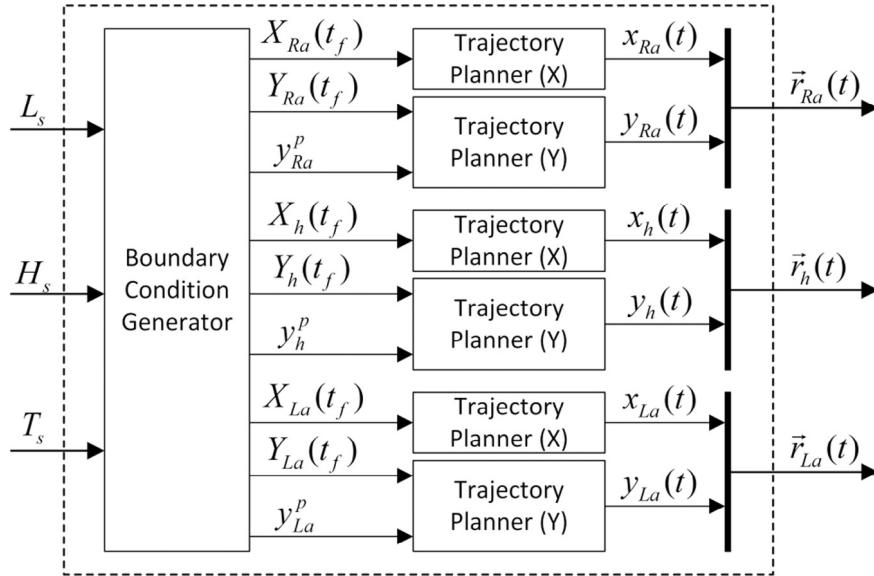


Fig. 4. Pattern generator block diagram.

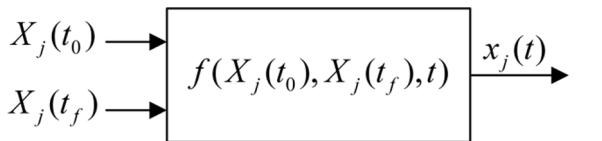


Fig. 5. The most common trajectory planner. \$X\_j(t\_0)\$ and \$X\_j(t\_f)\$ refer to the boundary conditions of the trajectory.

**Lemma 1.**  $u(t) = v_x(t)$  steers the states of the system  $S_x$  from the initial value  $X_j(t_0) = [x_1(t_0) \ x_2(t_0) \ x_3(t_0)]^T$  to the final value  $X_j(t_f) = [x_1(t_f) \ x_2(t_f) \ x_3(t_f)]^T$  in such a way that the cost function  $J_x$  has been minimized.  $v_x(t)$  is obtained as the following:

$$v_x(t) = a(t - t_0)^2 - b(t - t_0) + c \quad (7)$$

where

$$\begin{aligned} a &= \frac{360(x_1(t_f) - x_1(t_0))}{(t_f - t_0)^5} - \frac{180(x_2(t_f) + x_2(t_0))}{(t_f - t_0)^4} \\ &\quad + \frac{30(x_3(t_f) - x_3(t_0))}{(t_f - t_0)^3} \\ b &= \frac{360(x_1(t_f) - x_1(t_0))}{(t_f - t_0)^4} - \frac{24(7x_2(t_f) + 8x_2(t_0))}{(t_f - t_0)^3} \end{aligned}$$

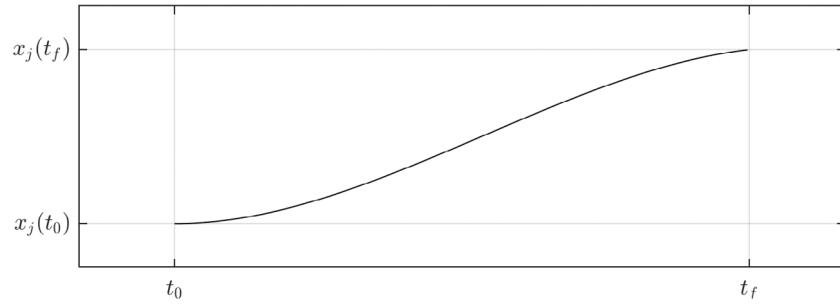
$$c = \frac{\frac{12(2x_3(t_f) - 3x_3(t_0))}{(t_f - t_0)^2}}{(t_f - t_0)^3} - \frac{12(2x_2(t_f) + 3x_2(t_0))}{(t_f - t_0)^2} + \frac{3(x_3(t_f) - 3x_3(t_0))}{(t_f - t_0)}. \quad (8)$$

**Proof.** According to the minimum principle of Pontryagin [17], minimization of  $J_x$  can be achieved by minimizing the Hamiltonian function defined as

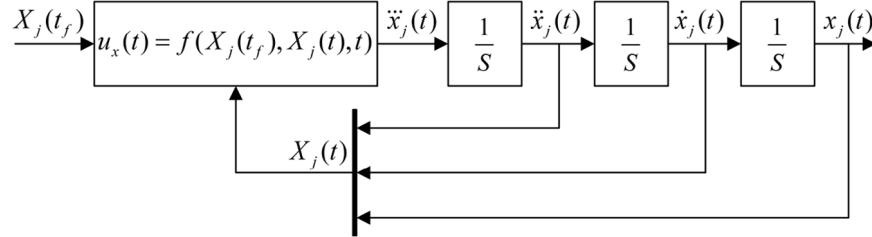
$$H(X_j(t), v_x(t), P(t)) = v_x^2(t) + p_1(t)x_2(t) + p_2(t)x_3(t) + p_3(t)v_x(t) \quad (9)$$

where  $P(t) = [p_1(t) \ p_2(t) \ p_3(t)]^T$  is defined as co-state vector. The optimal trajectories  $X_j^*(t)$  and  $v_x^*(t)$  can be achieved by satisfying the following conditions:

$$\begin{aligned} \dot{X}_j^*(t) &= \frac{\partial H}{\partial P}(X_j^*(t), v_x^*(t), P^*(t), t) \\ \dot{P}^*(t) &= -\frac{\partial H}{\partial X_j}(X_j^*(t), v_x^*(t), P^*(t), t) \\ 0 &= \frac{\partial H}{\partial v_x}(X_j^*(t), v_x^*(t), P^*(t), t) \end{aligned} \quad (10)$$



**Fig. 6.** A general trajectory shape for the x component of the position of the joints.



**Fig. 7.** The proposed trajectory planner for the x component of the position of the joints.

where the symbol \* refers to the extremals of  $X_j(t)$ ,  $v_x(t)$ , and  $P(t)$ . The necessary conditions for optimality can be written as

$$\begin{aligned} \dot{x}_1^*(t) &= x_2^*(t) \\ \dot{x}_2^*(t) &= x_3^*(t) \\ \dot{x}_3^*(t) &= -0.5p_3^*(t) \\ \dot{p}_1^*(t) &= 0 \\ \dot{p}_2^*(t) &= -p_1^*(t) \\ \dot{p}_3^*(t) &= -p_2^*(t) \\ v_x^*(t) &= -0.5p_3^*(t). \end{aligned} \quad (11)$$

By applying the initial condition  $X_j(t_0)$  and the final value  $X_j(t_f)$ , the optimal control function for  $t \in [t_0, t_f]$  is obtained as (7).

**Lemma 1** represents an open-loop control method to steer the states of the system  $S_x$  from a specific initial condition to a fixed final value along with minimizing  $J_x$ . In this control method,  $v_x(t)$  is determined in  $t = t_0$  for  $t \in [t_0, t_f]$  interval and the final value is not changeable in  $t \in (t_0, t_f]$ . Using a closed-loop control method and online calculation of  $u$  makes the final value changeable. For this purpose, **Theorem 1** is represented.

**Theorem 1.** Feedback control law  $u(t) = u_x(t)$  steers the states of the system  $S_x$  from any initial value to the final value  $X_j(t_f) = [x_1(t_f) \ x_2(t_f) \ x_3(t_f)]^T = [x_j(t_f) \ \dot{x}_j(t_f) \ \ddot{x}_j(t_f)]^T$ , in such a way that the cost function  $J_x$  is minimized.  $u_x(t)$  is obtained for  $t < t_f$  as the following:

$$\begin{aligned} u_x(t) &= \frac{60(x_j(t_f) - x_1(t))}{(t_f - t)^3} - \frac{12(2\dot{x}_j(t_f) + 3x_2(t))}{(t_f - t)^2} \\ &\quad + \frac{3(\ddot{x}_f - 3x_3(t))}{(t_f - t)}. \end{aligned} \quad (12)$$

**Proof.** Consider the state feedback  $X_j(t) = [x_1(t) \ x_2(t) \ x_3(t)]^T$  as the initial condition of the system  $S_x$  at any moment. The online calculation of  $u$  in respect to the new defined initial condition allows the final value  $X_j(t_f)$  to be changeable. Considering  $X_j(t)$  as  $X_j(t_0)$  corresponds to putting  $t$  instead of  $t_0$  in the formulation of  $v_x(t)$  in **Lemma 1**. In other words:

$$u_x(t) = \{v_x(t)|t_0 = t\} \quad (13)$$

Therefore, from **Lemma 1** and (13),  $u_x(t)$  is obtained as (12).

The trajectory obtained by applying  $u_x(t)$  to the system  $S_x$  is denoted by  $T_x(t, t_f, X_j(t_f))$ , where:

$$T_x(t, t_f, X_j(t_f)) = \{x_j(t)|u = u_x(t), t < t_f\}. \quad (14)$$

### 3.2. The y component of the joints

**Fig. 8** Shows a general trajectory shape for the y component of the position of the joints, rising from the initial condition  $Y_j(t_0) = [y_j(t_0) \ \dot{y}_j(t_0) \ \ddot{y}_j(t_0)]^T$  to a peak of  $y_j^p$ , and converging to  $Y_j(t_f) = [y_j(t_f) \ \dot{y}_j(t_f) \ \ddot{y}_j(t_f)]^T$ , with the minimum curvature.

We propose the feedback controlled third order system depicted in **Fig. 9** in order to plan this trajectory, which can be formulated as follows:

$$S_y: \begin{cases} \dot{y}_1 = y_2 \\ \dot{y}_2 = y_3 \\ \dot{y}_3 = u \\ y_j = y_1 \end{cases} \quad (15)$$

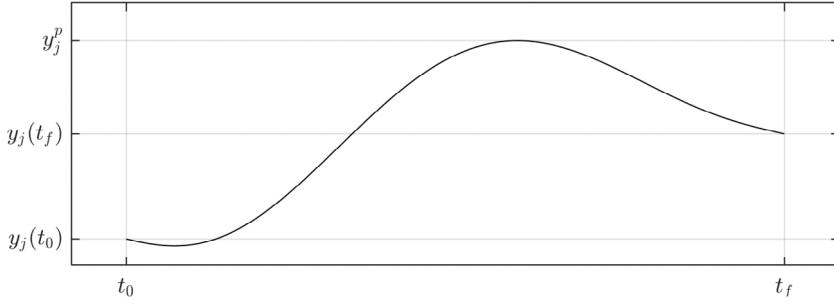
In order to increase the trajectory's smoothness, the cost function denoted by  $J_y$  is intended as (16). Minimizing  $J_y$  causes a peak on the trajectory in addition to the smoothness increment, wherein the parameter  $k$  determines the value of the peak.

$$J_y = \int_{t_0}^{t_f} \left( \frac{d^3 y_j}{dt^3} \right)^2 + k y_j(t) dt. \quad (16)$$

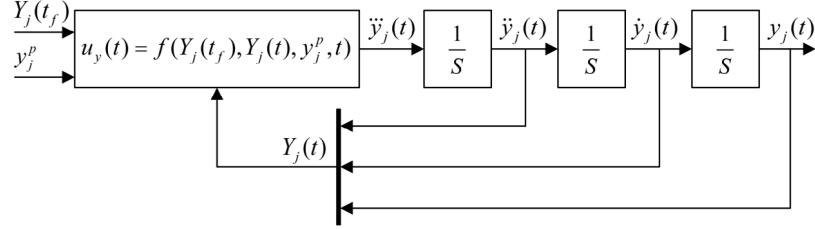
We have presented **Theorem 2** to design the proper feedback control law  $u$  in order to move the states of the system  $S_y$  to the final condition  $Y_j(t_f)$ , along with minimizing the cost function  $J_y$ . **Lemma 2** is used to prove of **Theorem 2**.

**Lemma 2.**  $u(t) = v_y(t)$  steers the states of the system  $S_y$  from the initial value  $Y_j(t_0) = [y_1(t_0) \ y_2(t_0) \ y_3(t_0)]^T$  to the final value  $Y_j(t_f) = [y_1(t_f) \ y_2(t_f) \ y_3(t_f)]^T$  in such a way that the cost function  $J_y$  is minimized.  $v_y(t)$  is obtained as the following:

$$v_y(t) = \frac{k}{12} (t - t_0)^3 + a(t - t_0)^2 - b(t - t_0) + c \quad (17)$$



**Fig. 8.** The general trajectory shape for the y component of the position of the joints.



**Fig. 9.** The proposed trajectory planner for the y component of the position of the joints.

where

$$\begin{aligned} a &= \frac{360(y_1(t_f) - y_1(t_0))}{(t_f - t_0)^5} - \frac{180(y_2(t_f) + y_2(t_0))}{(t_f - t_0)^4} \\ &\quad + \frac{30(y_3(t_f) - y_3(t_0))}{(t_f - t_0)^3} - \frac{k(t_f - t_0)}{8} \\ b &= \frac{360(y_1(t_f) - y_1(t_0))}{(t_f - t_0)^4} - \frac{24(7y_2(t_f) + 8y_2(t_0))}{(t_f - t_0)^3} \\ &\quad + \frac{12(2y_3(t_f) - 3y_3(t_0))}{(t_f - t_0)^2} - \frac{k(t_f - t_0)^2}{20} \\ c &= \frac{60(y_1(t_f) - y_1(t_0))}{(t_f - t_0)^3} - \frac{12(2y_2(t_f) + 3y_2(t_0))}{(t_f - t_0)^2} \\ &\quad + \frac{3(y_3(t_f) - 3y_3(t_0))}{(t_f - t_0)} - \frac{k(t_f - t_0)^3}{240}. \end{aligned} \quad (18)$$

**Proof.** As in proof of Lemma 1, the Hamiltonian function is defined as

$$\begin{aligned} H(Y_j(t), v_y(t), P(t)) &= v_y^2(t) + ky_1(t) + p_1(t)y_2(t) \\ &\quad + p_2(t)y_3(t) + p_3(t)v_y(t) \end{aligned} \quad (19)$$

The necessary conditions for optimality can be written as

$$\begin{aligned} \dot{y}_1^*(t) &= y_2^*(t) \\ \dot{y}_2^*(t) &= y_3^*(t) \\ \dot{y}_3^*(t) &= -0.5p_3^*(t) \\ \dot{p}_1^*(t) &= -k \\ \dot{p}_2^*(t) &= -p_1^*(t) \\ \dot{p}_3^*(t) &= -p_2^*(t) \\ v_y^*(t) &= -0.5p_3^*(t). \end{aligned} \quad (20)$$

By applying the initial condition  $Y_j(t_0)$  and the final value  $Y_j(t_f)$ , the optimal control function can be obtained for  $t \in [t_0, t_f]$  as (17).

**Theorem 2.** Feedback control law  $u(t) = u_y(t)$  steers the states of the system  $S_y$  from any initial value to the final value  $Y_j(t_f) = [y_1(t_f) \ y_2(t_f) \ y_3(t_f)]^T = [y_j(t_f) \ \dot{y}_j(t_f) \ \ddot{y}_j(t_f)]^T$ , in such a way

that the cost function  $J_y$  is minimized.  $u_y(t)$  is obtained for  $t < t_f$  as the following:

$$\begin{aligned} u_y(t) &= \frac{60(y_j(t_f) - y_1(t))}{(t_f - t)^3} - \frac{12(2\dot{y}_j(t_f) + 3y_2(t))}{(t_f - t)^2} \\ &\quad + \frac{3(\ddot{y}_j(t_f) - 3y_3(t))}{(t_f - t)} - \frac{K(t_f - t)^3}{240}. \end{aligned} \quad (21)$$

**Proof.** As in proof of Theorem 1:

$$u_y(t) = \{v_y(t)|t_0 = t\} \quad (22)$$

From Lemma 2 and (22),  $u_y(t)$  is obtained as (21).

**Calculation of  $k$ :**

According to the designed feedback control law (21) we have

$$\int_{t_0}^{t_f} y_j(t) dt \Big|_{k=k^*} - \int_{t_0}^{t_f} y_j(t) dt \Big|_{k=0} = -k^* \times \frac{(t_f - t_0)^7}{5 \times 8!} \quad (23)$$

where  $k^*$  denotes the proper value of  $k$  for generating a trajectory with a peak of  $y_j^p$ .

By describing the trajectories by piecewise fifth-order polynomials, the approximate integral of  $y_j(t)$  for  $k = 0$  and for  $k = k^*$  can be calculated numerically as

$$\int_{t_0}^{t_f} y_j(t) dt \Big|_{k=0} \cong \frac{1}{120} \begin{bmatrix} 60y_j(t_0) \\ 12\dot{y}_j(t_0) \\ \ddot{y}_j(t_0) \\ 60y_j(t_f) \\ -12\dot{y}_j(t_f) \\ \ddot{y}_j(t_f) \end{bmatrix}^T \cdot \begin{bmatrix} (t_f - t_0) \\ (t_f - t_0)^2 \\ (t_f - t_0)^3 \\ (t_f - t_0) \\ (t_f - t_0)^2 \\ (t_f - t_0)^3 \end{bmatrix} \quad (24)$$

$$\int_{t_0}^{t_f} y_j(t) dt \Big|_{k=k^*} \cong \frac{1}{120} \begin{bmatrix} 60y_j(t_0) \\ 12\dot{y}_j(t_0) \\ \ddot{y}_j(t_0) \\ 60y_j^p \\ 0 \\ 0 \end{bmatrix}^T \cdot \begin{bmatrix} (t_p - t_0) \\ (t_p - t_0)^2 \\ (t_p - t_0)^3 \\ (t_p - t_0) \\ (t_p - t_0)^2 \\ (t_p - t_0)^3 \end{bmatrix} \quad (25)$$

$$+ \frac{1}{120} \begin{bmatrix} 60y_j^p \\ 0 \\ 0 \\ 60y_j(t_f) \\ -12\dot{y}_j(t_f) \\ \ddot{y}_j(t_f) \end{bmatrix}^T \cdot \begin{bmatrix} (t_f - t_p) \\ (t_f - t_p)^2 \\ (t_f - t_p)^3 \\ (t_f - t_p) \\ (t_f - t_p)^2 \\ (t_f - t_p)^3 \end{bmatrix}. \quad (25)$$

where the parameter  $t_p$  refers to the peak time and is obtained approximately by

$$t_p \cong t_0 + \frac{y_j^p - y_j(t_0)}{2y_j^p - y_j(t_0) - y_j(t_f)}(t_f - t_0) \quad (26)$$

Using (23), (24), and (25) the value of  $k^*$  can be determined.

The trajectory obtained by applying  $u_y(t)$  to the system  $S_y$  is denoted by  $T_y(t, t_0, t_f, Y_j(t_0), Y_j(t_f), y_j^p)$ , where:

$$T_y(t, t_0, t_f, Y_j(t_0), Y_j(t_f), y_j^p) = \{y_j(t) | u = u_y(t), t < t_f\}. \quad (27)$$

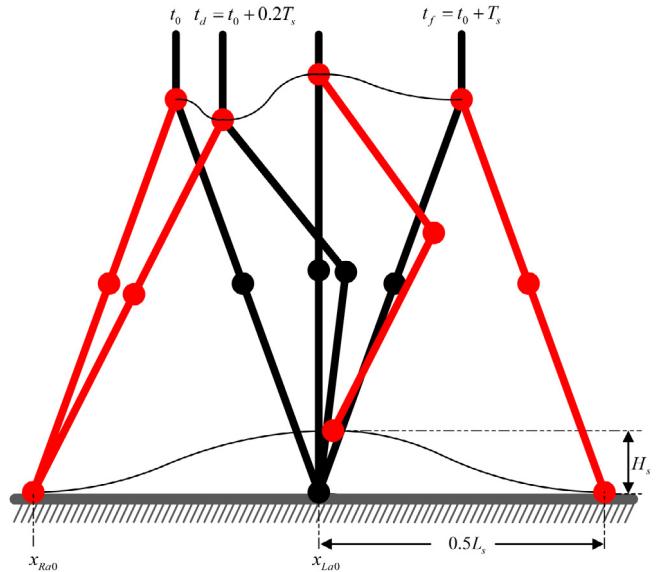
$T_x$  and  $T_y$  are employed as real-time smooth trajectory planners for planning the trajectory of the x and y component of the position of each joint respectively. The only parameters needed for planning the trajectories are  $(t_f, X_j(t_f))$  for  $T_x$ , and  $(t_0, t_f, Y_j(t_0), Y_j(t_f), y_j^p)$  for  $T_y$ , which can be extracted as the endpoint boundary conditions of the trajectories.

In Section 4, the endpoint boundary conditions are calculated for each joint in respect to walking constraints and to prevent a backward balance loss. For this purpose, the gait cycle is divided into two phases: the double support phase and the single support phase. The boundary conditions of the double support phase are determined to avoid a backward balance loss; and the endpoint boundary conditions are calculated for the single support phase in order to satisfy the walking parameters and constraints.

#### 4. Endpoint boundary conditions

In medical lower limb exoskeleton, the patient uses parallel bars, walkers or crutches as he/she walks. When using parallel bars, the stability is guaranteed in both forward and backward direction. But for the case of using walkers or crutches, the walking pattern should be able to avoid a backward balance loss. The well-known ZMP stability criterion cannot be used confidently for the motion planning because the disturbance coming from the user has a significant effect on the zero moment point. By modeling the robot as an inverted pendulum, the body center of mass (CoM) is an alternative index for assessing the stability. To prevent falling backward at the beginning of the single support phase, the velocity of the CoM must be greater than a critical value. For this purpose, a double support phase (approximately 20% of the gait cycle) is required in order to increase the velocity from zero to the critical value. Fig. 10 demonstrates the fundamental parameters involved in a stride cycle beginning from the initial ground contact position denoted by  $x_{Ra0}$  and  $x_{La0}$ . In this stride, the right leg (Red) swings while the left one (Black) stays on the ground. The robot's posture is displayed at different moments demonstrating the walking constraints, which are given as

$$\begin{aligned} r_L(t) &= l^2 \text{ for } t = t_0, t_0 + T_s \\ r_R(t) &= l^2 \text{ for } t = t_0, t_d, t_0 + T_s \\ \max(y_h) &= l \\ \max(y_{Ra}) &= H_s \end{aligned} \quad (28)$$



**Fig. 10.** Parameters description in a stride cycle . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

where  $l = l_s + l_r$ . The parameters  $r_L(t)$  and  $r_R(t)$  are defined as

$$r_L(t) = (x_h(t) - x_{La}(t))^2 + (y_h(t) - y_{La}(t))^2 \quad (29)$$

In order to generating walking pattern, the boundary conditions of the joints should be determined for each phase (double support phase:  $t_0 \leq t < t_0 + 0.2T_s$ , single support phase:  $t_0 + 0.2T_s \leq t < t_0 + T_s$ ).

##### 4.1. Double support phase

At the end of the double support phase, by ignoring the effect of the crutches, the robot can be regarded as an inverted pendulum at  $t = t_d$  as shown in Fig. 11. For the sake of simplicity, the effect of patient interaction with robot is considered in a simple way. In fact, as the patient is paraplegic and his/her legs do not apply torque, it is a good approximation to model it just as mass and moment of inertia. Assuming that the CoM is approximately located on the hip, the following constraint should be satisfied to avoid a backward balance loss at the beginning of the single support phase [16]:

$$\dot{x}_h(t_d) > \omega(x_{La0} - x_h(t_d)), \quad \omega = \sqrt{g/l} \quad (30)$$

where  $\omega$  is the natural frequency of the pendulum. On the other hand, it can be estimated that

$$x_h(t_d) - x_h(t_0) \approx 0.1T_s \times \dot{x}_h(t_d). \quad (31)$$

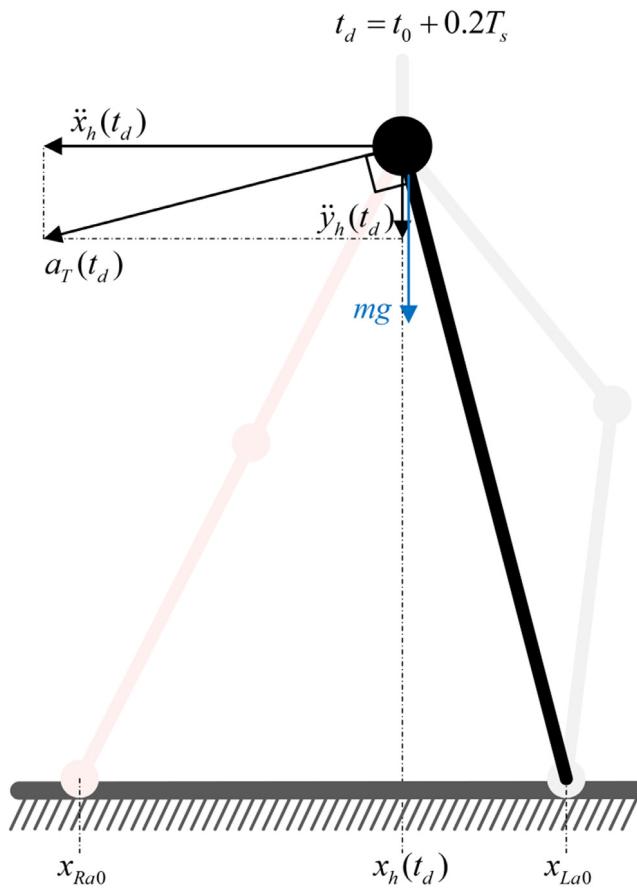
Using (30) and (31), the position and the minimum velocity of the hip in the horizontal direction at  $t = t_d$  can be determined as

$$x_h(t_d) = \frac{x_{Ra0} + x_{La0}(1 + 0.2\omega T_s)}{2 + 0.2\omega T_s} \quad (32)$$

$$\dot{x}_h(t_d) = \omega(x_{La0} - x_h(t_d)). \quad (33)$$

Regarding to Fig. 11, by using Euler's equation, the tangential acceleration of the center of mass at  $t = t_d$  can be obtained as

$$a_T(t_d) = \frac{g(x_{La0} - x_h(t_d))}{\sqrt{(x_{La0} - x_h(t_d))^2 + y_h^2(t_d)}}. \quad (34)$$



**Fig. 11.** The inverted pendulum model of the robot at  $t = t_d$ .

By setting the radial component of the center of mass acceleration to zero at  $t = t_d$ , the value of  $\ddot{x}_h(t_d)$  and  $\ddot{y}_h(t_d)$  can be defined as

$$\ddot{x}_h(t_d) = -\frac{g(x_{La0} - x_h(t_d)) \times y_h(t_d)}{(x_{La0} - x_h(t_d))^2 + y_h^2(t_d)} \quad (35)$$

$$\ddot{y}_h(t_d) = -\frac{g(x_{La0} - x_h(t_d))^2}{(x_{La0} - x_h(t_d))^2 + y_h^2(t_d)}. \quad (36)$$

The following constraints should be imposed to prevent an ill-posed inverse kinematics problem caused by an excessive distance between the hip and the right ankle.

$$r_R(t_d) = l^2 \quad (37)$$

$$\dot{r}_R(t_d) = 0 \quad (38)$$

$$\ddot{r}_R(t_d) < 0. \quad (39)$$

Under the above constraints and using (32), (33), (35), and (36), the boundary conditions of the x and y components of each joint's position at the end of the double support phase can be obtained as in Tables 1 and 2. Furthermore, the inequality (39) can be rewritten as

$$\ddot{x}_h(t_d)(x_h(t_d) - x_{Ra0}) + \ddot{y}_h(t_d)y_h(t_d) + \dot{x}_h^2(t_d) + \dot{y}_h^2(t_d) < 0. \quad (40)$$

It can be proved that the defined boundary conditions satisfy the inequality (40).

#### 4.2. Single support phase

In the single support phase, the right leg swings from  $x_{Ra0}$  to  $x_{La0} + 0.5L_s$ , while the left leg rests on  $x_{La0}$ . Regarding to the

**Table 1**

Boundary conditions of the x component of the joints position for the double support phase ( $t_f = t_0 + 0.2T_s$ ).

Joint	Hip	Left ankle	Right ankle
$x_j(t_f)$	$\frac{x_{Ra0} + x_{La0}(1+0.2\omega T_s)}{2+0.2\omega T_s}$	$x_{La0}$	$x_{Ra0}$
$\dot{x}_j(t_f)$	$\omega(x_{La0} - x_h(t_f))$	0	0
$\ddot{x}_j(t_f)$	$-\frac{g(x_{La0} - x_h(t_f)) \times y_h(t_f)}{(x_{La0} - x_h(t_f))^2 + y_h^2(t_f)}$	0	0

**Table 2**

Boundary conditions of the y component of each joints position for the double support phase ( $t_f = t_0 + 0.2T_s$ ).

Joint	Hip	Left ankle	Right ankle
$y_j(t_f)$	$\sqrt{l^2 - (x_h(t_f) - x_{Ra0})^2}$	0	0
$\dot{y}_j(t_f)$	$\frac{x_{Ra0} - x_h(t_f)}{y_h(t_f)} \times \dot{x}_h(t_f)$	0	0
$\ddot{y}_j(t_f)$	$-\frac{g(x_{La0} - x_h(t_f))^2}{(x_{La0} - x_h(t_f))^2 + y_h^2(t_f)}$	0	0
$y_j^p$	$\sqrt{l^2 - 0.25(x_{La0} - x_{Ra0})^2}$	0	0

**Table 3**

Boundary conditions of the x component of the joints position for the single support phase ( $t_f = t_0 + T_s$ ).

Joint	Hip	Left ankle	Right ankle
$x_j(t_f)$	$x_{La0} + 0.25L_s$	$x_{La0}$	$x_{La0} + 0.5L_s$
$\dot{x}_j(t_f)$	0	0	0
$\ddot{x}_j(t_f)$	0	0	0

constraints (28):

$$x_h(t_0 + T_s) = x_{La0} + 0.25L_s \quad (41)$$

$$y_h(t_0 + T_s) = \sqrt{l^2 - (0.25L_s)^2}. \quad (42)$$

In addition, the following constraints should be imposed to prevent an ill-posed inverse kinematics problem caused by an excessive distance between the hip and the ankle joints

$$\dot{r}_L(t_0 + T_s) = 0 \quad (43)$$

$$\dot{r}_R(t_0 + T_s) = 0 \quad (44)$$

$$\ddot{r}_L(t_0 + T_s) < 0 \quad (45)$$

$$\ddot{r}_R(t_0 + T_s) < 0. \quad (46)$$

Given that  $\dot{x}_{La}(t_0 + T_s) = \dot{x}_{Ra}(t_0 + T_s) = \dot{y}_{La}(t_0 + T_s) = \dot{y}_{Ra}(t_0 + T_s) = 0$ , and using (43) and (44), it can be shown that

$$\dot{x}_h(t_0 + T_s) = 0 \quad (47)$$

$$\dot{y}_h(t_0 + T_s) = 0. \quad (48)$$

Furthermore, by setting  $\ddot{x}_{La}(t_0 + T_s) = \ddot{x}_{Ra}(t_0 + T_s) = \ddot{y}_{La}(t_0 + T_s) = \ddot{y}_{Ra}(t_0 + T_s) = 0$ , the inequalities (45) and (46) can be rewritten as

$$\ddot{y}_h(t_0 + T_s) < -\frac{L_s}{4y_h(t_0 + T_s)} |\ddot{x}_h(t_0 + T_s)|. \quad (49)$$

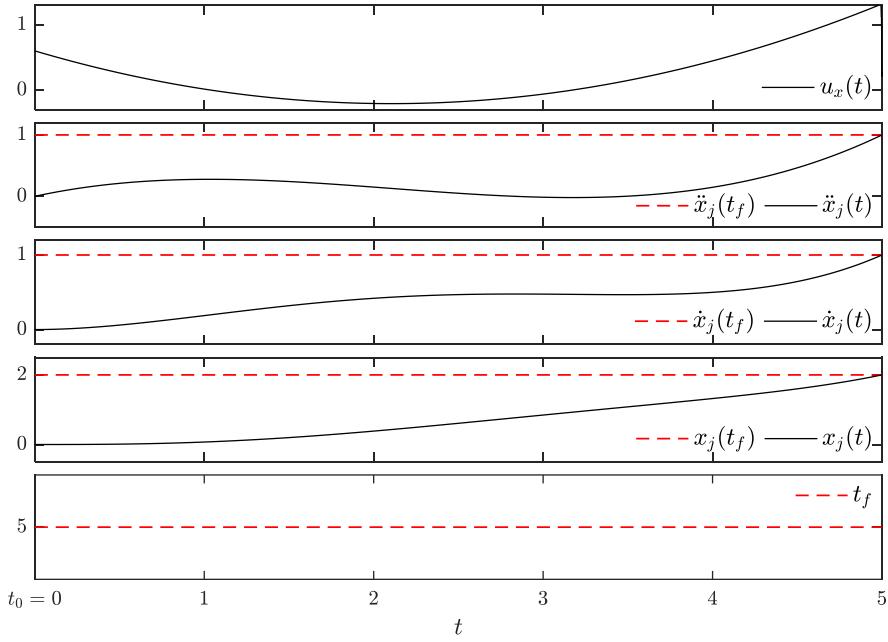
A proper choice for  $\ddot{x}_h$  and  $\ddot{y}_h$  is proposed as

$$\ddot{x}_h(t_0 + T_s) = 0 \quad (50)$$

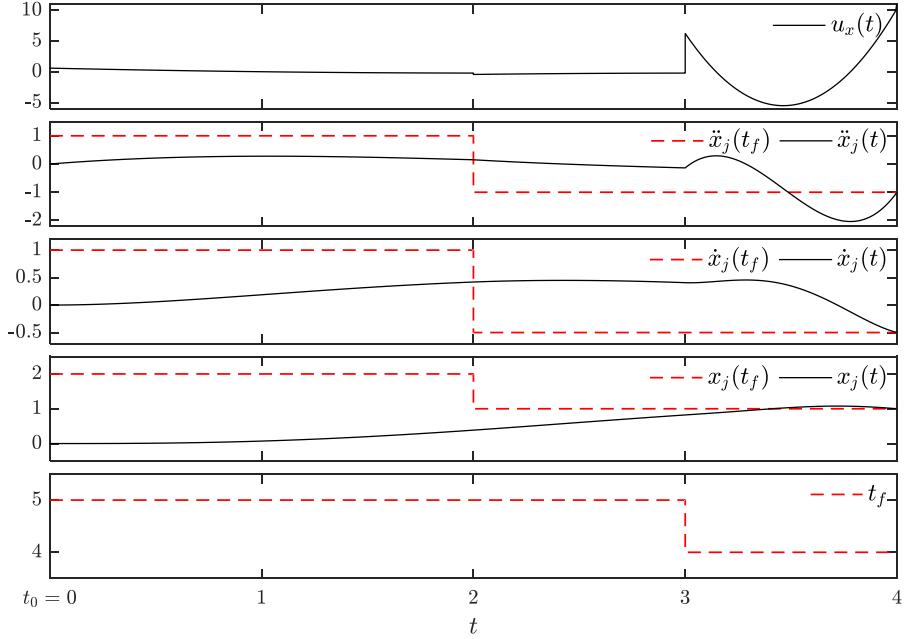
$$\ddot{y}_h(t_0 + T_s) = -\frac{2L_s}{T_s^2}. \quad (51)$$

The boundary conditions of the joints' positions for the single support phase are summarized in Tables 3 and 4.

The real-time walking pattern generation method is developed using the proposed real-time trajectory planners  $T_x$  and  $T_y$ , and the determined endpoint boundary conditions. By calculating  $x_R$ ,



**Fig. 12.** Trajectory planned by  $T_x$  for fixed boundary conditions. Dashed lines indicate the boundary conditions. Solid lines indicate the planned trajectory, the first and the second derivatives, and the control signal.



**Fig. 13.** Trajectory planned by  $T_x$  for changeable boundary conditions. Dashed lines indicate the boundary conditions. Solid lines indicate the planned trajectory, the first and the second derivatives, and the control signal.

**Table 4**

Boundary conditions of the  $y$  component of each joints position for the single support phase ( $t_f = t_0 + T_s$ ).

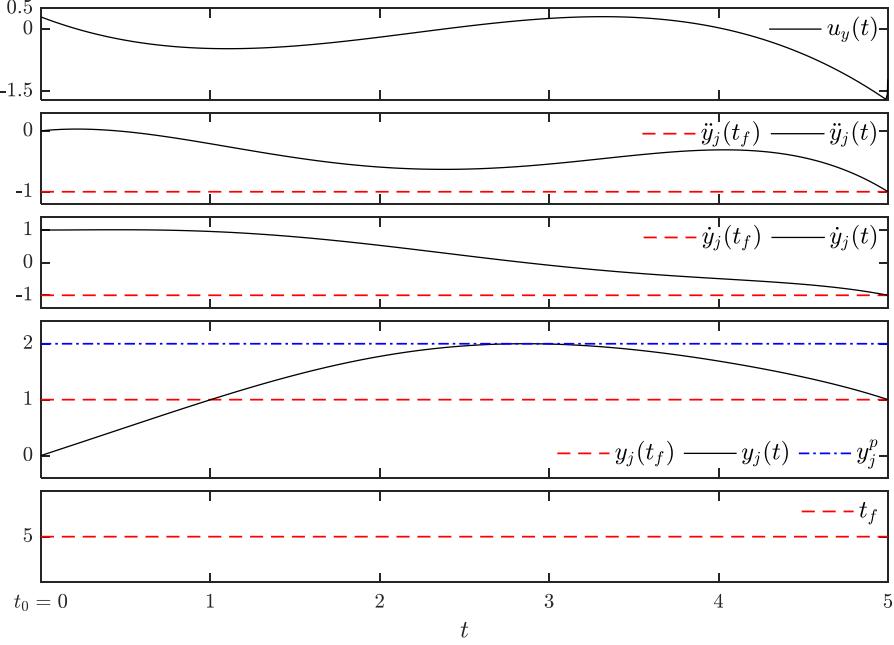
Joint	Hip	Left ankle	Right ankle
$y_j(t_f)$	$\sqrt{l^2 - (0.25L_s)^2}$	0	0
$\dot{y}_j(t_f)$	0	0	0
$\ddot{y}_j(t_f)$	$-\frac{2l_s}{T_s^2}$	0	0
$y_j^p$	$l$	0	$H_s$

$y_R$ ,  $x_L$ , and  $y_L$  from (2) and applying inverse kinematics given by (3), the joint angles will be obtained.

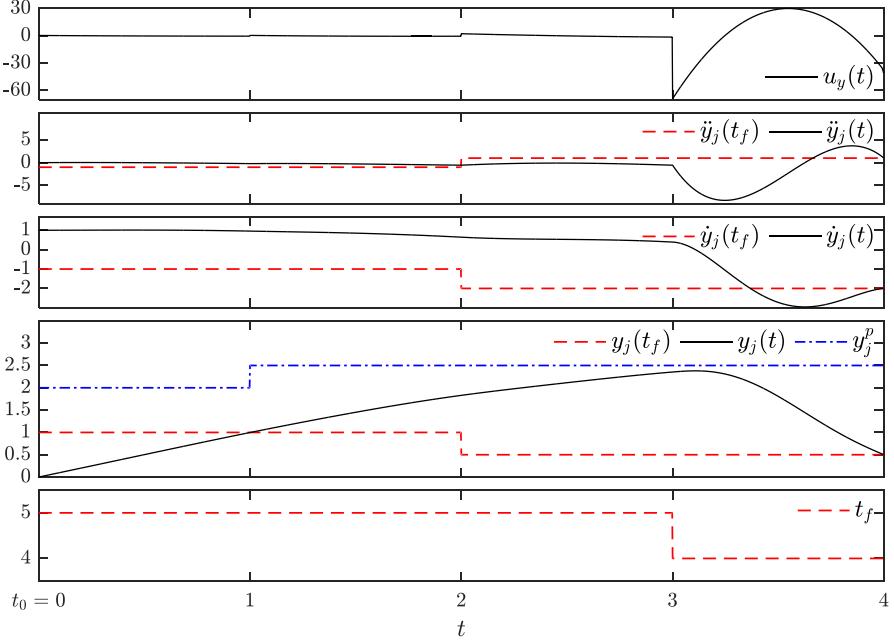
## 5. Experimental results

### 5.1. Simulation of the designed trajectory planners

The designed trajectory planners  $T_x$  and  $T_y$  play the main roles in the walking pattern generation and have an effect on the performance of the gait. Therefore a performance analysis of the trajectory planners is required, especially for their response to the changing boundary conditions. Resulting from the simulation, Figs. 12 and 13 show the trajectories planned by  $T_x$ , for fixed and changeable boundary conditions respectively. Fig. 12 shows the generated trajectory starting from the initial condition  $X_j(t_0) = [0 \ 0 \ 0]^T$  and ending up at the endpoint boundary condition



**Fig. 14.** Trajectory planned by  $T_y$  for fixed boundary conditions. Dashed lines indicate the boundary conditions. Dot-dashed line indicates the peak value. Solid lines indicate the planned trajectory, the first and the second derivatives, and the control signal.



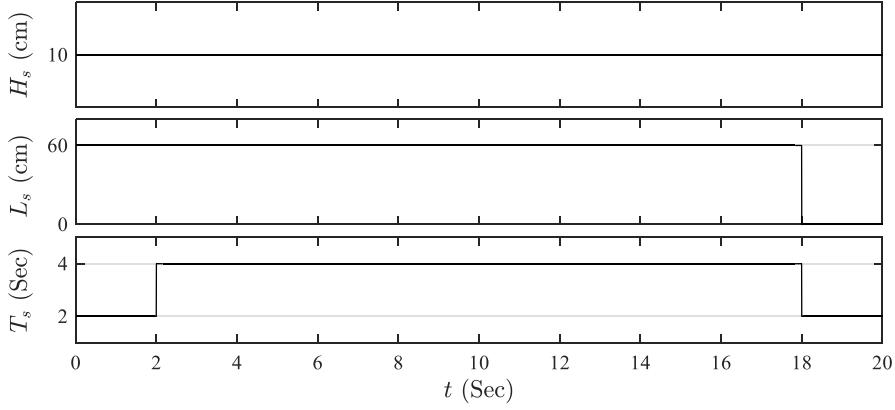
**Fig. 15.** Trajectory planned by  $T_y$  for changeable boundary conditions. Dashed lines indicate the boundary conditions. Dot-dashed line indicates the peak value. Solid lines indicate the planned trajectory, the first and the second derivatives, and the control signal.

$X_j(t_f) = [2 \ 1 \ 1]^T$  at  $t_f = 5$ . While in Fig. 13, the boundary conditions change from  $X_j(t_f) = [2 \ 1 \ 1]^T$  to  $X_j(t_f) = [1 \ -0.5 \ -1]^T$  at  $t = 2$  and the boundary time is brought forward from  $t_f = 5$  to  $t_f = 4$  at  $t = 3$ . Similarly for  $T_y$ , Figs. 14 and 15 represent the trajectories, respectively for fixed and changeable boundary conditions. Fig. 14 shows the trajectory generated by  $T_y$  for  $Y_j(t_0) = [0 \ 1 \ 0]^T$ ,  $Y_j(t_f) = [1 \ -1 \ -1]^T$ ,  $y_j^p = 2$ , and  $t_f = 5$ . While in Fig. 15, the boundary parameters change three times at  $t = 1, 2$ , and  $3$ . In the figures, the dashed lines are the boundary conditions, while the solid lines indicate the planned trajectory, the first and the second derivatives of the trajectory, and the control signal of the trajectory planner systems. As shown in

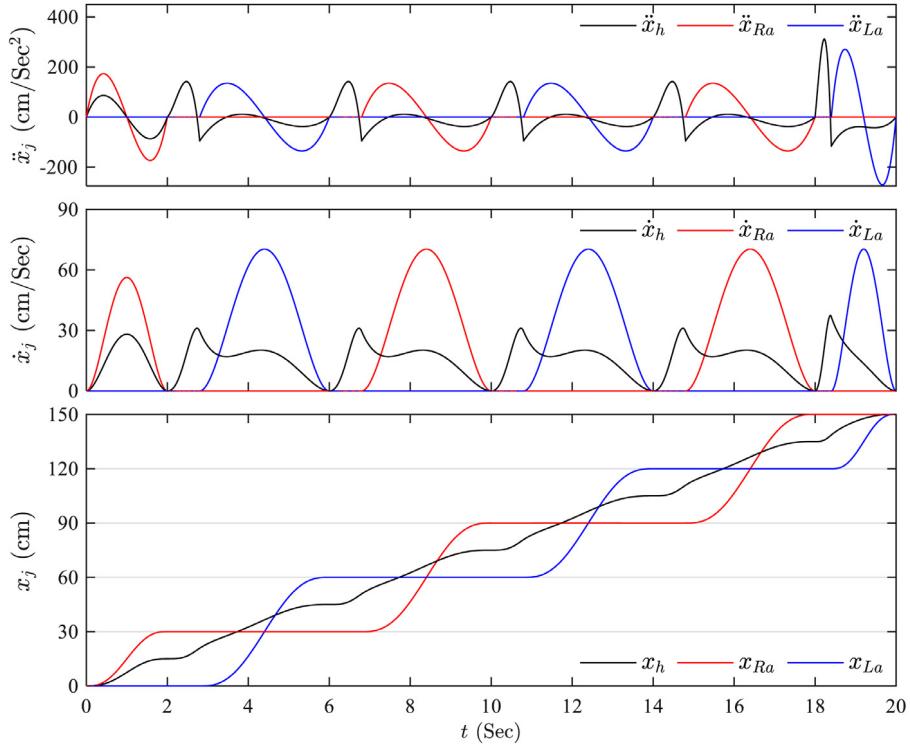
the figures, the continuity of the first and the second derivatives are maintained against the change of the parameters. As well as, the final conditions of the trajectories reach to the required boundaries at  $t = t_f$ . The results illustrate the good performance of the trajectory planners.

### 5.2. The real-time walking pattern implementation

For evaluating the proposed walking pattern generation method, different experiments with different walking parameters were carried out on the Exoped robot. In order to evaluate the robustness of the method across different users, each experiment



**Fig. 16.** Walking parameters of the first experiment.



**Fig. 17.** The x component of the position of the joints in the first experiment.

**Table 5**  
The characteristics of the participants.

Participant	$l_T$	$l_S$	Weight
User1	45 cm	51 cm	82 kg
User2	42 cm	47 cm	71 kg

was repeated by two participants described in Table 5. The experimental results of user1 are presented in the following. As well as, the effect of the participant's dimensions on the backward balance of the robot is studied by comparing the stability results of the users.

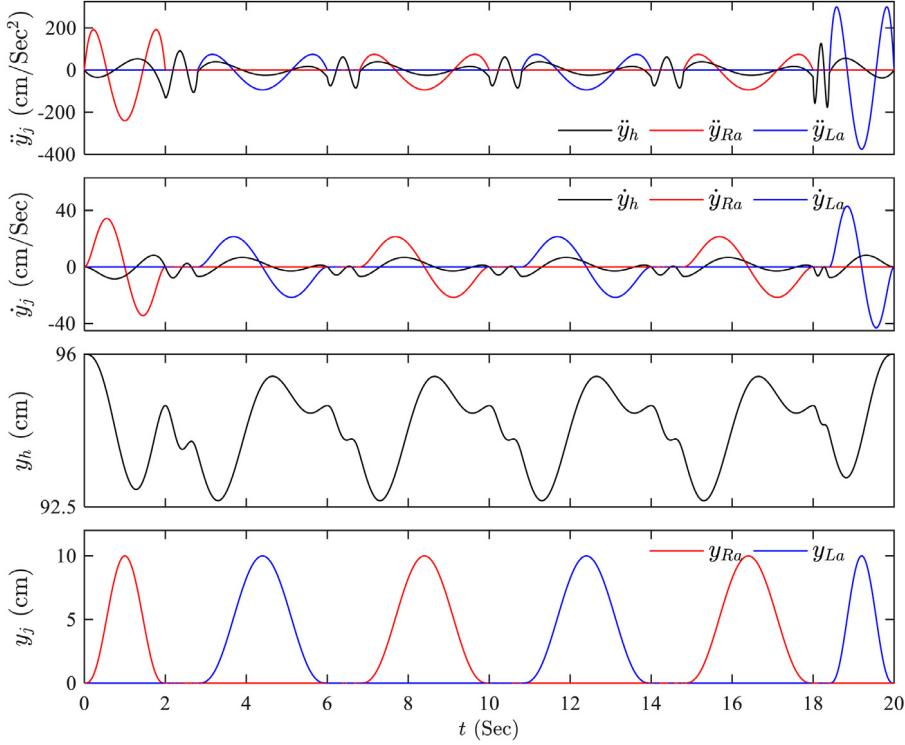
### 5.2.1. The 1st experiment

In the first experiment, walking begins from a standing pose and the right leg takes the first half-step. After taking four strides, the walking ends with a half-step taken by the left leg. Fig. 16 shows the walking parameters related to the first experiment, which took 20 s. As shown, the half-step and full-step time was 2 and 4 s respectively. Maximum foot clearance and step length were considered as  $H_s = 10$  cm and  $L_s = 60$  cm. The zero step length from  $t = 18$  s to  $t = 20$  s corresponds to the final half-step.

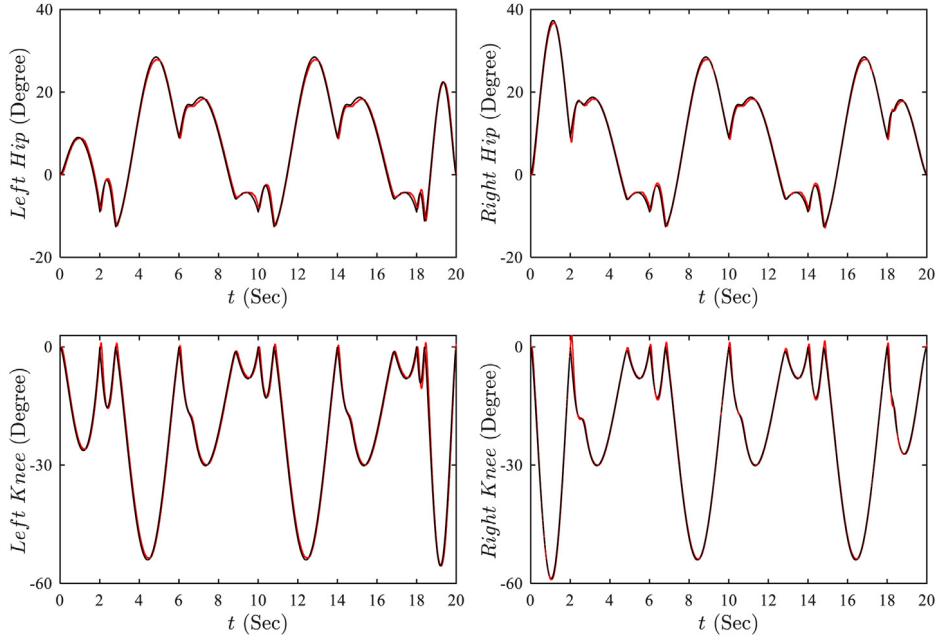
the walking ends with a half-step taken by the left leg. Fig. 16 shows the walking parameters related to the first experiment, which took 20 s. As shown, the half-step and full-step time was 2 and 4 s respectively. Maximum foot clearance and step length were considered as  $H_s = 10$  cm and  $L_s = 60$  cm. The zero step length from  $t = 18$  s to  $t = 20$  s corresponds to the final half-step.

Figs. 17 and 18 show the real-time position of the joints generated by the proposed method for the first experiment. As shown by the figures, the required walking parameters were satisfied. Moreover, all of the trajectories had a continuous second derivative.

By applying inverse kinematics transformations, the desired angle trajectories of the joints were obtained in real-time as shown in Fig. 19. A PID controller was used for regulation of



**Fig. 18.** The y component of the position of the joints in the first experiment.



**Fig. 19.** Experimental result; The reference angles (black lines) and the real angles (red lines) of the joints in the first experiment . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

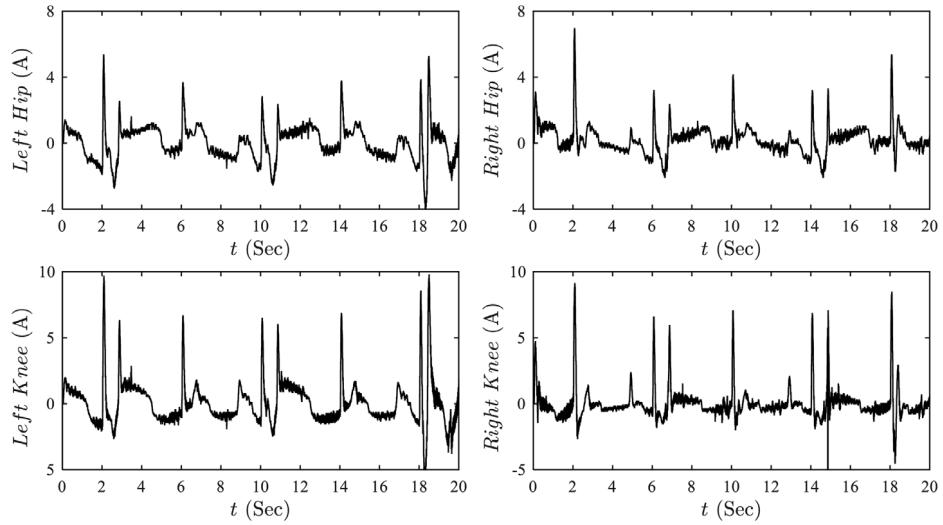
each motor's reference input. The obtained reference angles of the joints compared to the real angles measured by the hall sensors are depicted in the plots. Also, the motor current corresponding to each joint is shown in Fig. 20.

Fig. 21 Presents snapshots of whole walking cycle during the first experiment.

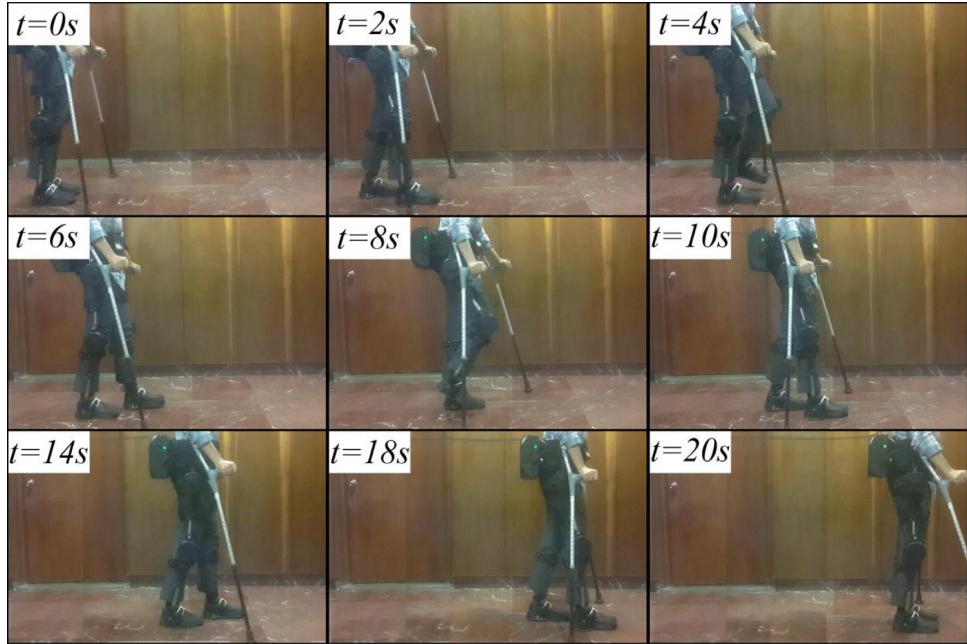
### 5.2.2. The 2nd experiment

The second experiment was carried out to evaluate the efficacy of the proposed method in obstacle crossing. As shown in Fig. 22, maximum foot clearance increased at  $t = 5$  s from  $H_s = 8$  cm to  $H_s = 15$  cm.

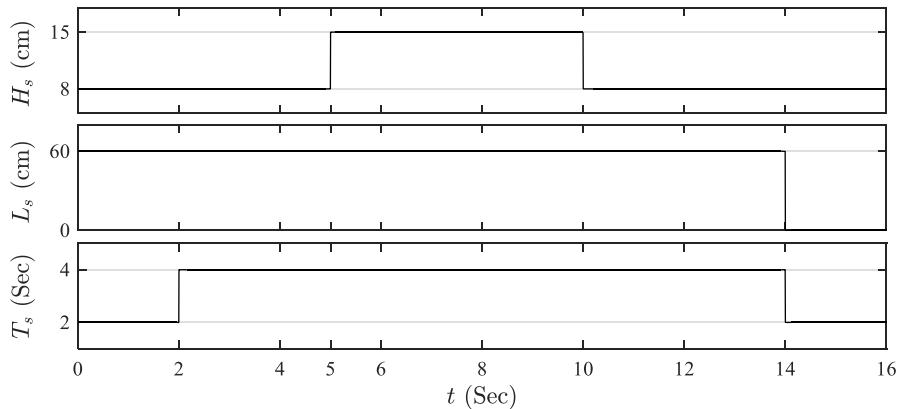
Figs. 23 and 24 show the generated real-time position of the joints. As shown in Fig. 24, the height of the stride increased from



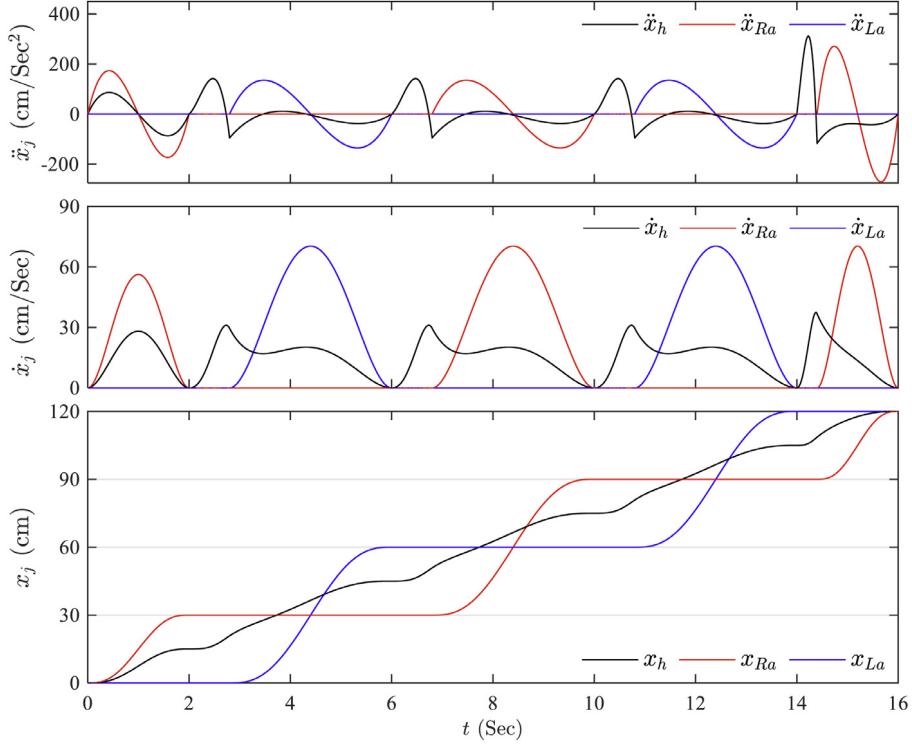
**Fig. 20.** Experimental result; The motor currents of the joints in the first experiment.



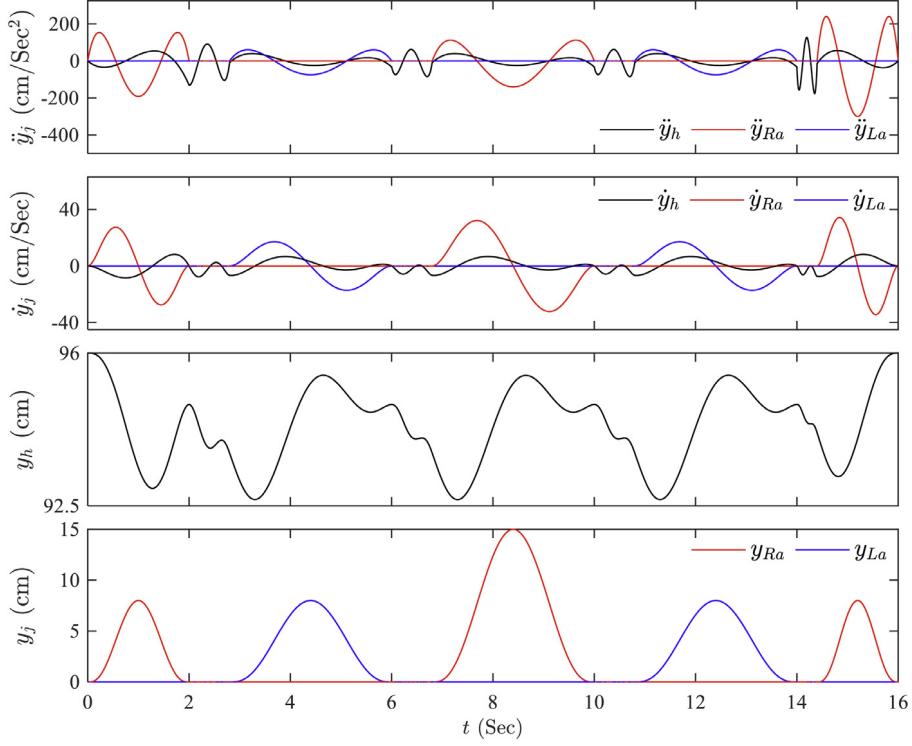
**Fig. 21.** Snapshots of whole walking cycle during the first experiment.



**Fig. 22.** Walking parameters of the 2nd experiment.



**Fig. 23.** The x component of the position of the joints in the 2nd experiment.

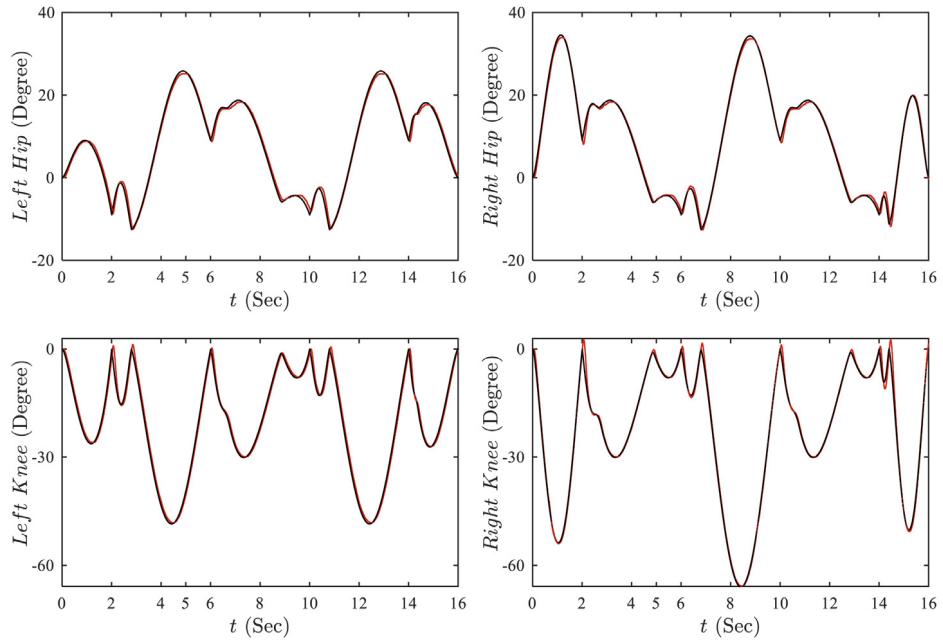


**Fig. 24.** The y component of the position of the joints in the 2nd experiment.

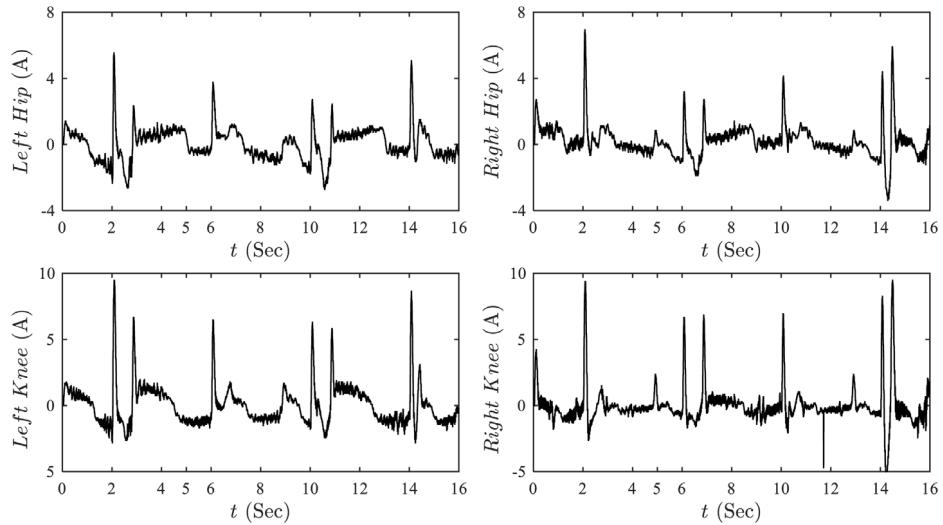
8 cm to 15 cm in the second full-step. In addition, the continuity of the second derivative of the trajectories are maintained in respect to the change of the parameters at  $t = 5$  s and  $t = 10$  s. Figs. 25 and 26 show the implementation results of the second experiment.

### 5.2.3. The 3rd experiment

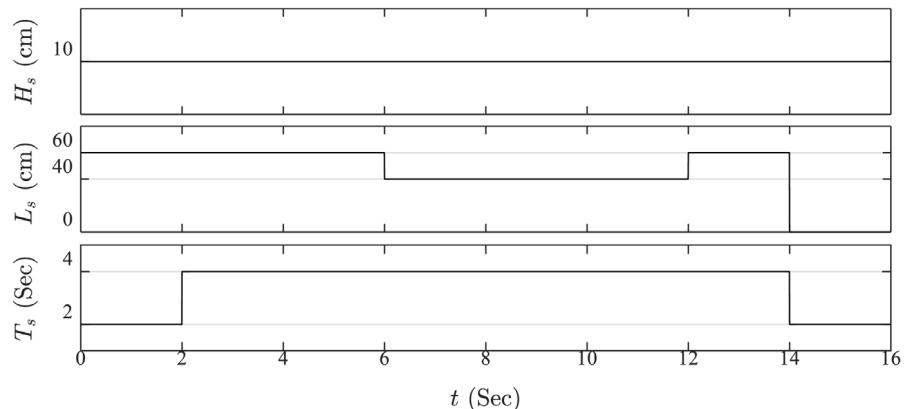
In the 3rd experiment the step length was shortened at the beginning of the second full-step and then reset back to the previous value at the middle of the third step. Fig. 27 shows the walking parameters during the experiment.



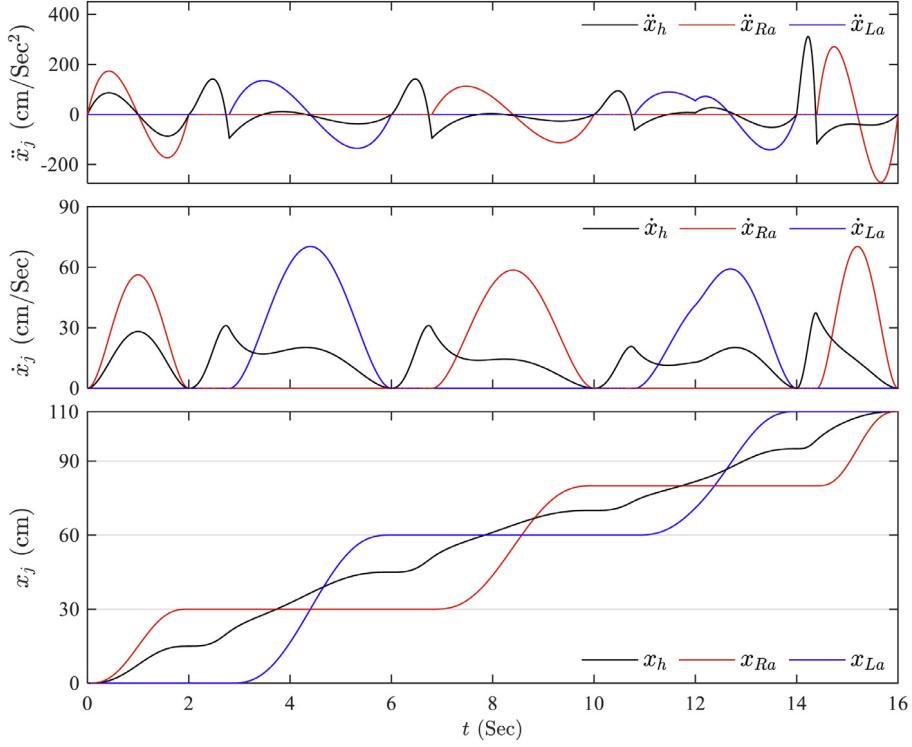
**Fig. 25.** Experimental result; The reference angles (black lines) and the real angles (red lines) of the joints in the 2nd experiment . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



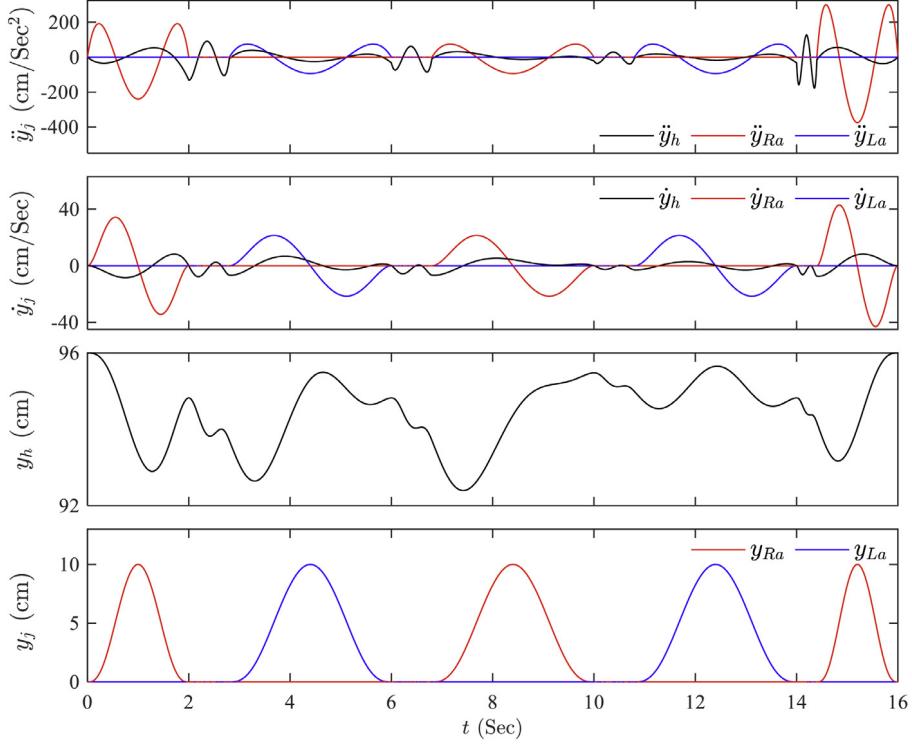
**Fig. 26.** Experimental result; The motor currents of the joints in the 2nd experiment.



**Fig. 27.** Walking parameters of the 3rd experiment.



**Fig. 28.** The x component of the position of the joints in the 3rd experiment.

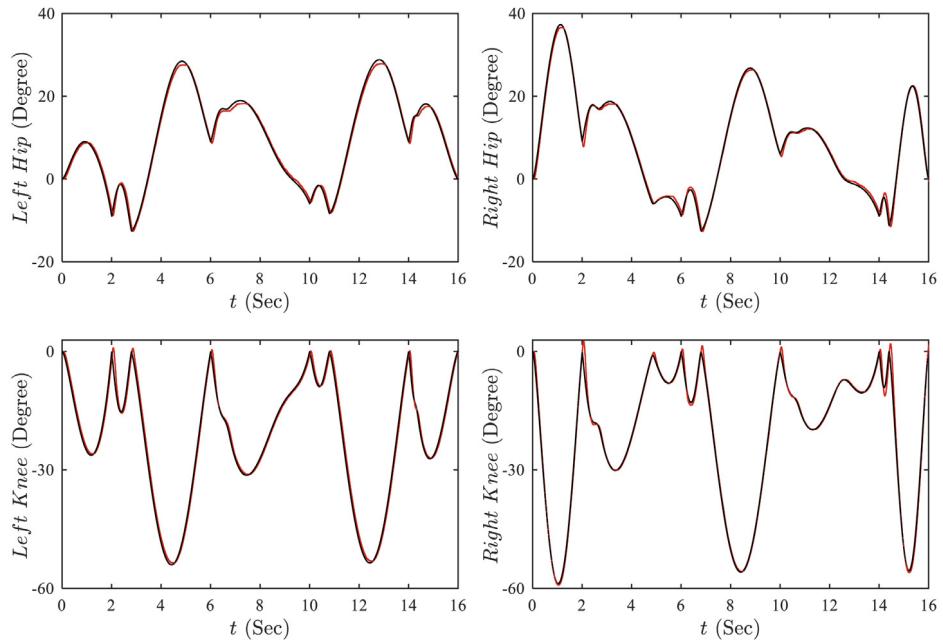


**Fig. 29.** The y component of the position of the joints in the 3rd experiment.

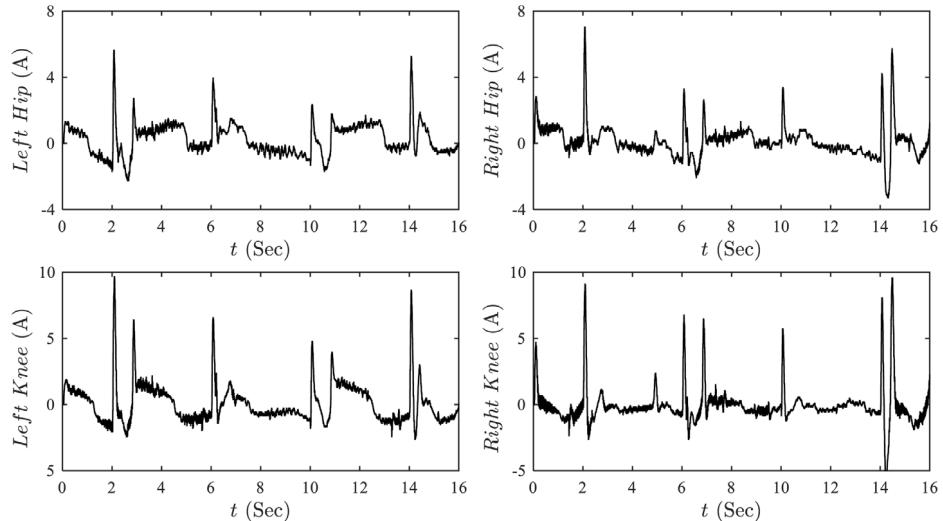
The generated real-time positions of the joints are shown in Figs. 28 and 29. As shown by the figures, the required step length was satisfied in addition to maintaining continuity of the second derivatives. Figs. 30 and 31 show the implementation results of the experiment.

#### 5.2.4. The 4th experiment

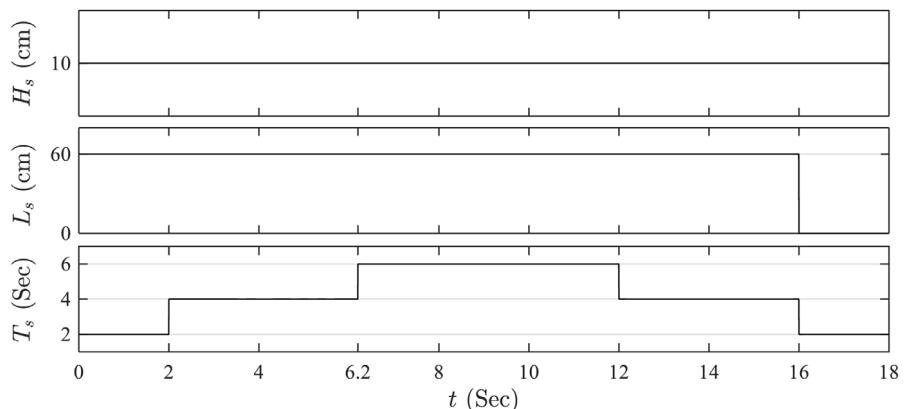
The real-time changing of the walking speed is implemented in the 4th experiment. As shown in Fig. 32 the walking speed decreased at  $t = 6.2$  s and then reset back to the previous value at  $t = 12$  s.



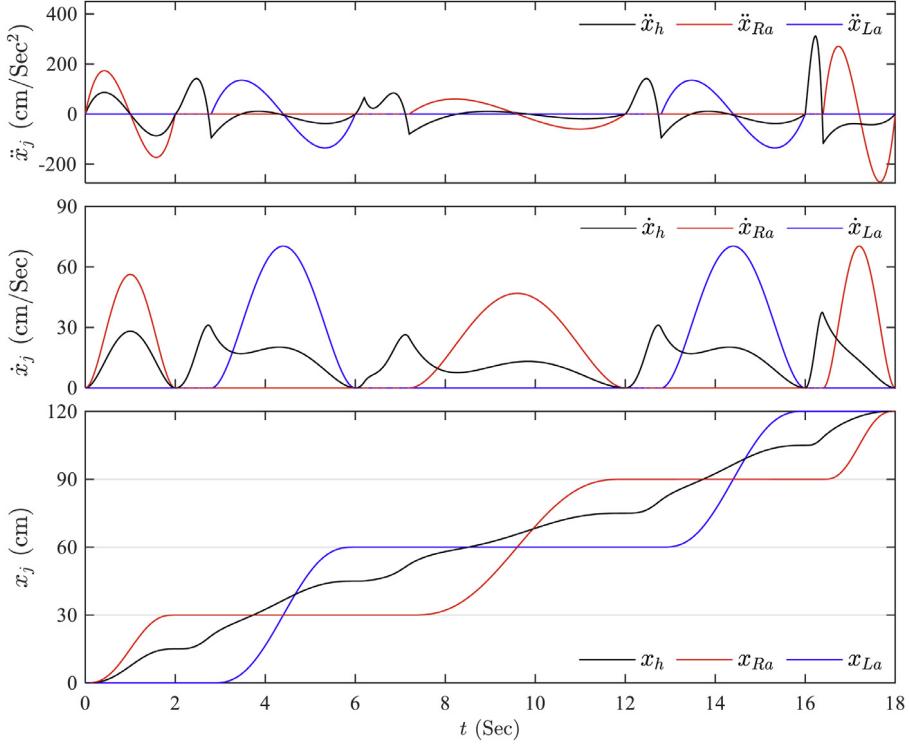
**Fig. 30.** Experimental result; The reference angles (black lines) and the real angles (red lines) of the joints in the 3rd experiment . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



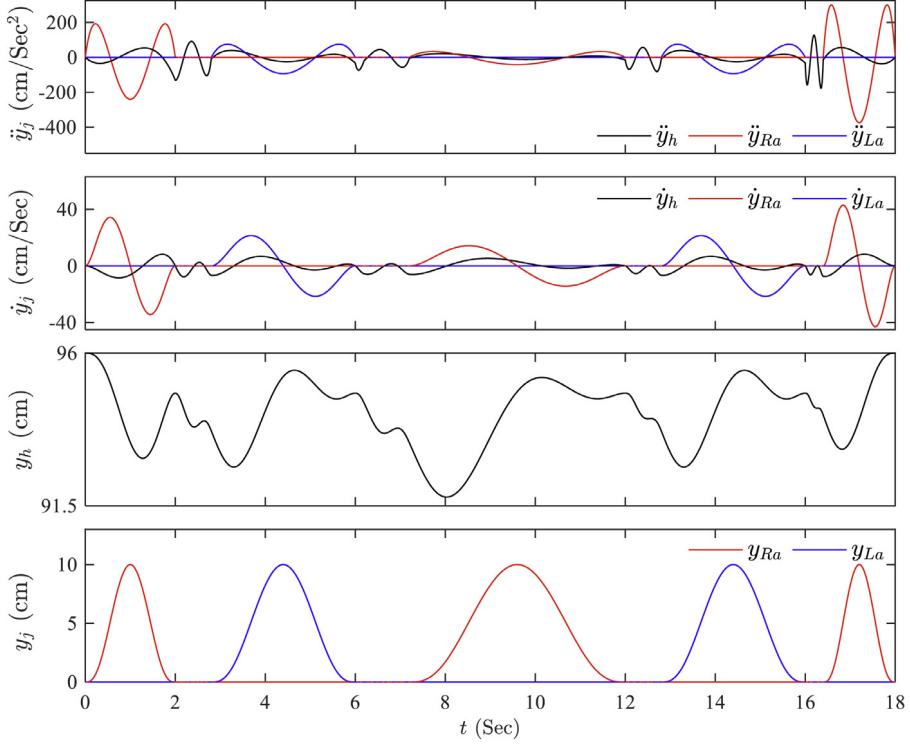
**Fig. 31.** Experimental result; The motor currents of the joints in the 3rd experiment.



**Fig. 32.** Walking parameters of the 4th experiment.



**Fig. 33.** The x component of the position of the joints in the 4th experiment.

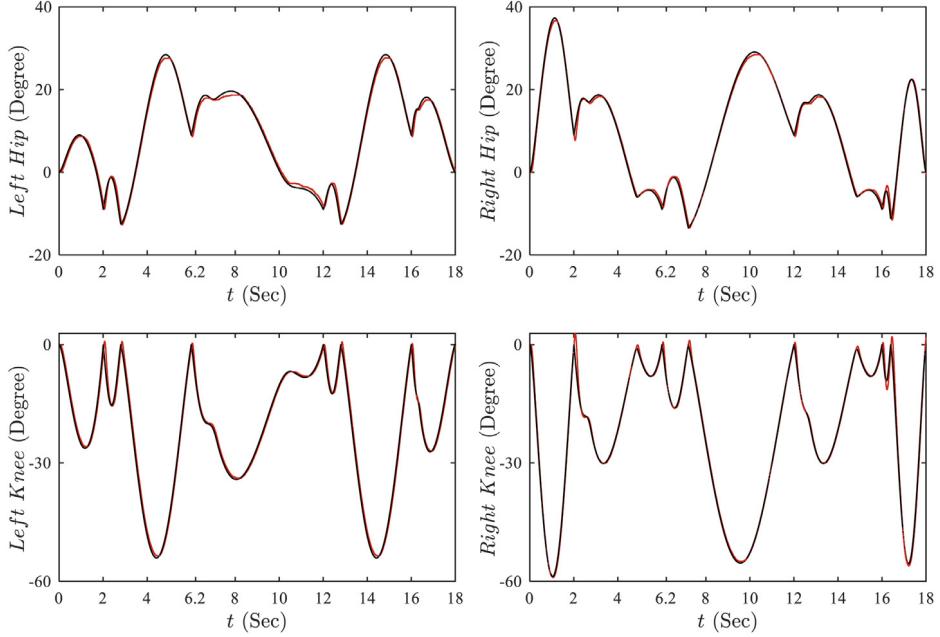


**Fig. 34.** The y component of the position of the joints in the 5th experiment.

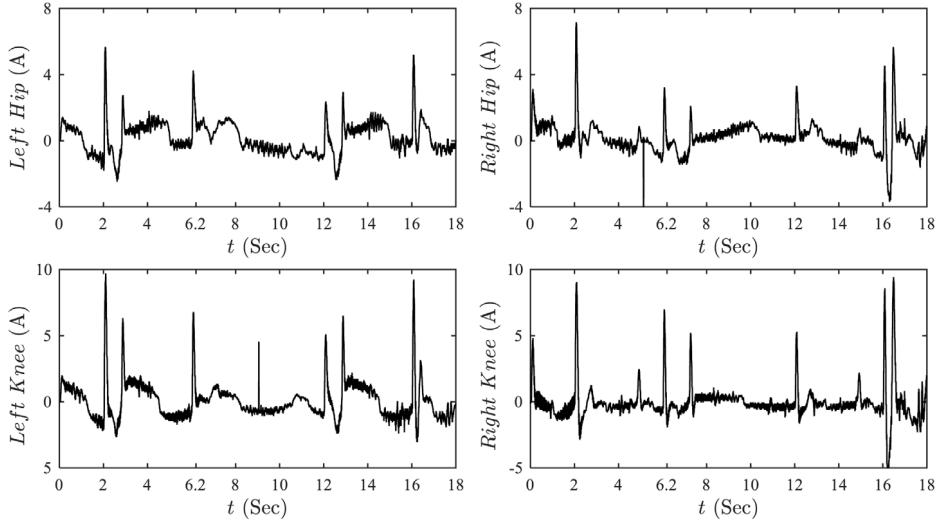
**Figs. 33 and 34** show the generated real-time position of the joints. As shown by the figures, the required walking parameters were satisfied in addition to maintaining continuity of the second derivatives. **Figs. 35 and 36** show the implementation results of the 4th experiment.

#### 5.2.5. The 5th experiment

The 5th experiment was carried out to evaluate the efficacy of the proposed method in response to the changes in all the parameters. **Fig. 37** shows the walking parameters during the experiment. As shown, maximum foot clearance and step length



**Fig. 35.** Experimental result; The reference angles (black lines) and the real angles (red lines) of the joints in the 4th experiment . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 36.** Experimental result; The motor currents of the joints in the 4th experiment.

**Table 6**

Comparing the stability of the gaits for user1 and user2.

Experiment	Participant	$\int SM(t)dt$	$\overline{SM}(t_0 + 0.2T_s)$
Exp1	User1	5.029	0.0312
	User2	5.139	0.0326
Exp2	User1	3.981	0.0314
	User2	4.067	0.0330
Exp3	User1	3.534	0.0290
	User2	3.609	0.0302
Exp4	User1	4.254	0.0278
	User2	4.338	0.0288
Exp5	User1	3.005	0.0279
	User2	3.066	0.0291

increased at  $t = 1.8$  s and  $t = 6$  s respectively. Also, walking speed decreased at  $t = 4.5$  s. The walking ends with a

decision to put the left leg down 20 cm ahead of the right leg in 1.2 s. This corresponds to setting  $L_s = 40$  cm and  $T_s = 2.7$  s.

Figs. 38 and 39 show the generated real-time position of the joints. As shown by the figures, the required walking parameters were satisfied in addition to maintaining continuity of the second derivatives. Figs. 40 and 41 show the implementation results of the 5th experiment.

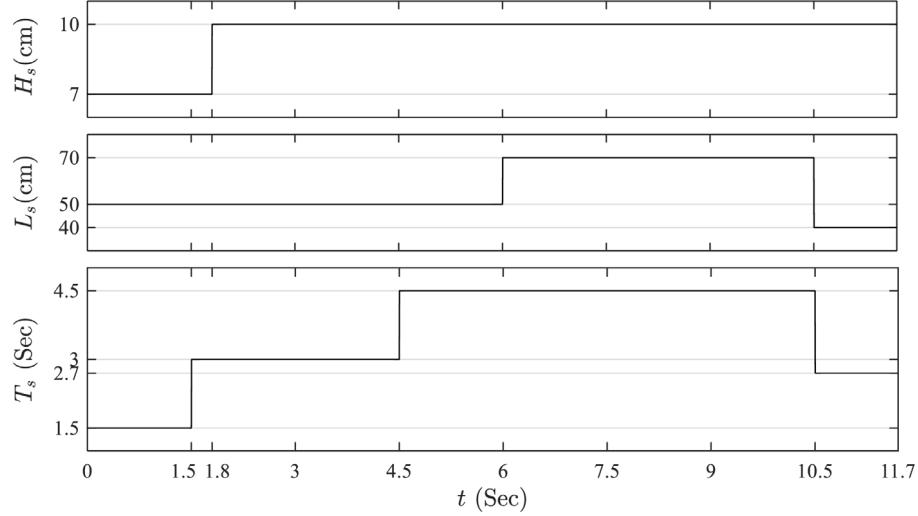
#### 5.2.6. Stability analysis

By defining the stability margin  $SM$  in the single support phase as

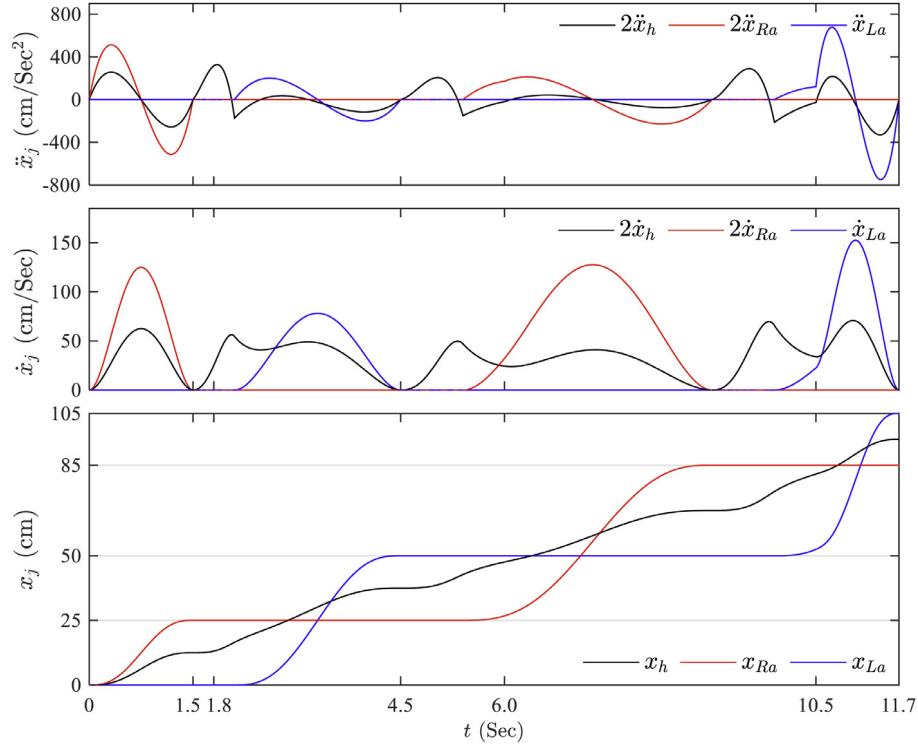
$$SM(t) = \dot{x}_h(t) - \omega(\max(x_{Ra}(t), x_{La}(t)) - x_h(t)), \quad (52)$$

the following constraint should be satisfied to avoid a backward balance loss during the single support phase

$$SM(t) > 0. \quad (53)$$



**Fig. 37.** Walking parameters of the 5th experiment.



**Fig. 38.** The x component of the position of the joints in the 5th experiment.

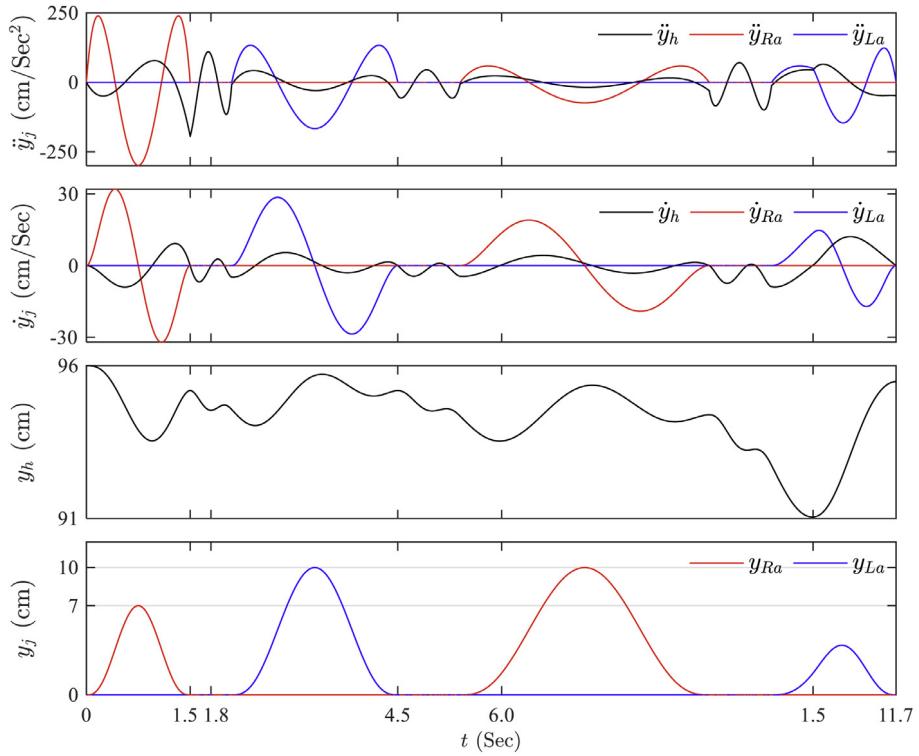
The value of  $SM(t)$  for each experiment is given in Fig. 42, in which  $SM(t)$  is positive for all experiments and the stability of the gaits are preserved. By comparing  $SM(t)$  in the first and second experiment, it can be said that in the proposed method the backward balance is not affected by  $H_s$ . While, the 4th experiment ( $6.2 < t < 12$ ) demonstrates that a significant decrease in walking speed can lead to backward balance loss.

The stability margin obtained for user1 and user2 is compared in order to investigate the effect of the user dimensions on the stability of the gaits. For this purpose, the integrals of  $SM(t)$  during the experiments, and the mean values of  $SM(t_0 + 0.2T_s)$  at the start time of the double support phases are computed. The obtained values are summarized in Table 6. By comparing the

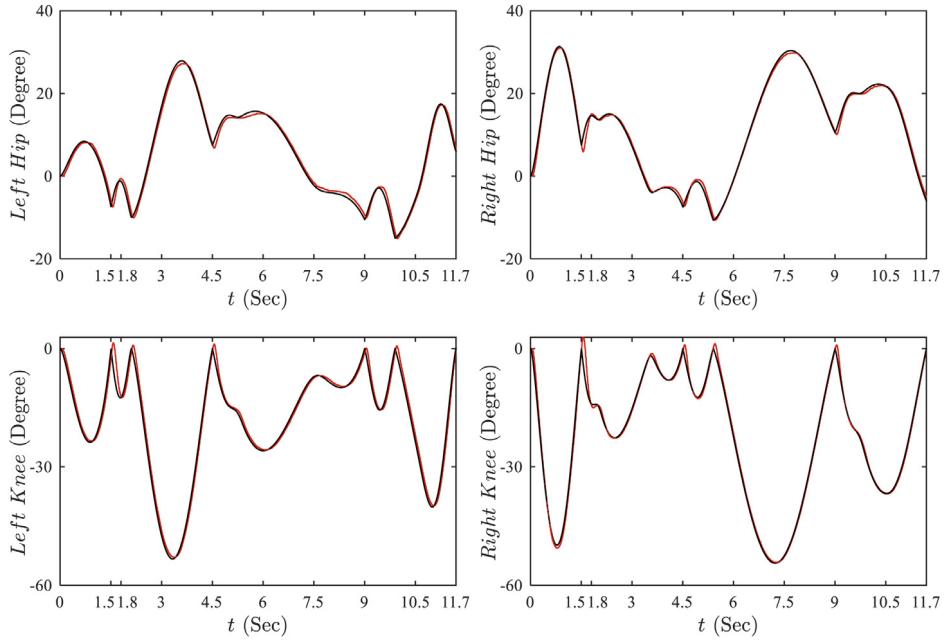
results for user1 and user2, it can be seen that the  $\sim 7\%$  decrease in  $l = l_T + l_S$  leads to  $\sim 4\%$  improvement in stability of the gaits at the critical time (beginning of the double support phases) and  $\sim 2\%$  improvement in stability of the gaits during the walking cycles.

## 6. Conclusion

A walking pattern generation method is proposed which generates the joint angles based on gait parameters. The provided pattern generator is designed in such a way that enables changing the gait parameters during the stride. For this purpose, the real-time trajectory planning problem is formulated as an optimal



**Fig. 39.** The y component of the position of the joints in the 5th experiment.

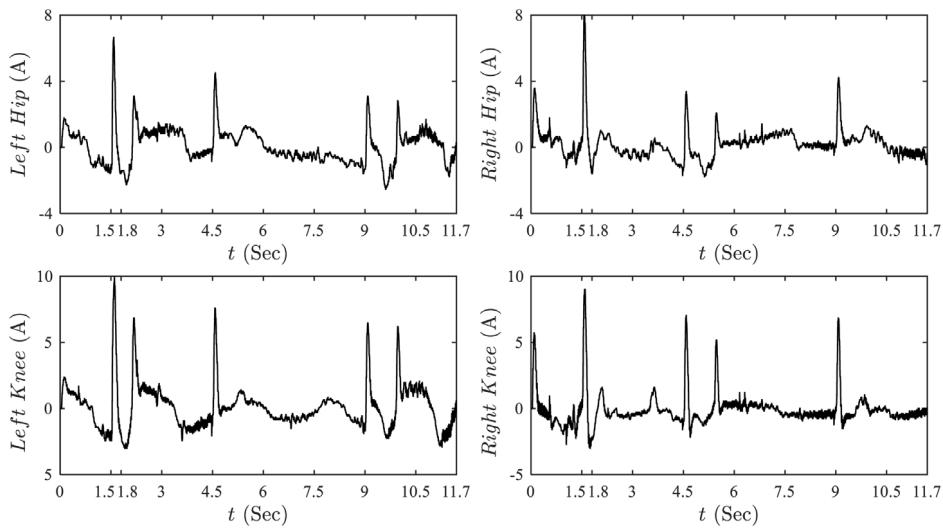


**Fig. 40.** Experimental result; The reference angles (black lines) and the real angles (red lines) of the joints in the 5th experiment . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

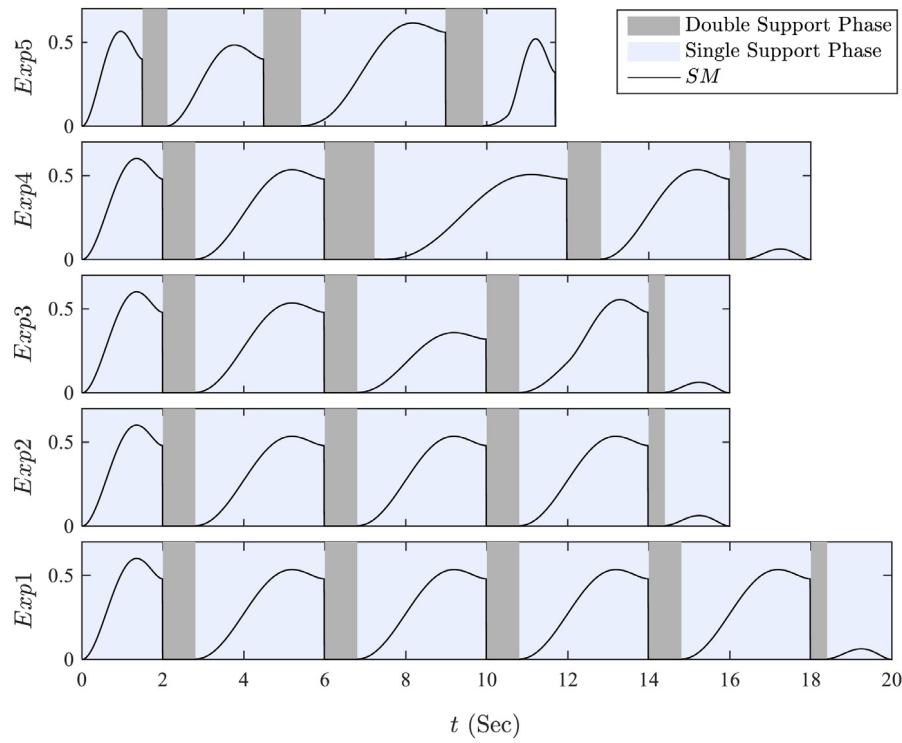
control problem with changeable final states; and a novel method is proposed to solve this problem. The trajectories generated by the proposed method have continuous second derivatives, even with the changing parameters of gait. In order to achieve a stable walking gait, the trajectory of the center of mass is defined in such a way to prevent the backward balance loss. Two experiments were carried out to evaluate the proposed method on the

Exopod robot. The results indicate that the robot has a smooth and stable motion even by changing the walking parameters.

The proposed method provides a natural walking pattern for the lower limb exoskeletons, which is adaptable to the change of the walking parameters in different situations and moments. Moreover, the proposed method provides a safe way for walking with crutches by maintaining the backward balance. Considering



**Fig. 41.** Experimental result; The motor currents of the joints in the 5th experiment.



**Fig. 42.** The value of  $SM(t)$  during the experiments for user1.  $SM(t)$  is defined as zero in double support phases.

the stability in the gait can help the user by reducing the force applied to the arms, especially in the clinical training between the parallel bars. According to Fig. 3 our future work is to design the walking parameters generator block. As an example, the sensor based determination of the parameters can be used for automating the movement of the robot in order to overcome obstacles and difficult situations. As well as, the real-time computing of the walking parameters according to the stability of the robot can make the movement of the robot safer. For this purpose, the foot force stability margin [19] is a simple and reliable index for assessing stability of a exoskeleton robot. In any case, the real-time determination of the walking parameters will be a potential application of the proposed walking pattern generator.

## References

- [1] R.S. Mosher, Handyman to Hardiman, *SAE Trans.* 76 (1968) 588–597.
- [2] A. Formalskii, A. Shneider, Book Reviews : Biped Locomotion (Dynamics Stability Control and Application): by M. Vukobratovich, B. Borovac, D. Surla, and D. Stokich published by Springer-Verlag, 1990, *Int. J. Robot. Res.* 11 (4) (1992) 396.
- [3] E. Guizzo, H. Goldstein, The rise of the body bots [robotic exoskeletons], *IEEE Spectr.* 42 (10) (2005) 50–56.
- [4] T. Yan, M. Cempini, C.M. Oddo, N. Vitiello, Review of assistive strategies in powered lower-limb orthoses and exoskeletons, *Robot. Auton. Syst.* 64 (2015) 120–136.
- [5] M. Vukobratović, B. Borovac, Zero-moment point – thirty five years of its life, *Int. J. Humanoid Robot.* 01 (01) (2004) 157–173.
- [6] H.F.N. Al-Shuka, B. Corves, W.-H. Zhu, B. Vanderborght, Multi-level control of zero-moment point-based humanoid biped robots: a review, *Robotica* 34 (11) (2016) 2440–2466.

- [7] H. Kazerooni, J.L. Racine, L. Huang, R. Steger, On the Control of the Berkeley Lower Extremity Exoskeleton (BLEEX), in: Proceedings of the IEEE 2005 International Conference on Robotics and Automation, 2005, pp. 4353–4360.
- [8] C. Lee, J.-Y. Kim, S.-Y. Kim, S. Oh, Human force observation and assistance for lower limb rehabilitation using wire-driven series elastic actuator, *Mechatronics* 55 (2018) 13–26.
- [9] A. Esquenazi, M. Talaty, A. Packel, M. Saulino, The rewalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury, *Am. J. Phys. Med. Rehabil.* 91 (11) (2012) 911–921.
- [10] K.A. Strausser, H. Kazerooni, The development and testing of a human machine interface for a mobile medical exoskeleton, in: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2011, pp. 4911–4916.
- [11] D. Sanz-Merodio, M. Cestari, J.C. Arevalo, E. Garcia, Control motion approach of a lower limb orthosis to reduce energy consumption, *Int. J. Adv. Robot. Syst.* 9 (6) (2012) 232.
- [12] J. Yoon, R.P. Kumar, A. Özer, An adaptive foot device for increased gait and postural stability in lower limb Orthoses and exoskeletons, *Int. J. Control Autom. Syst.* 9 (3) (2011) 515.
- [13] R. Huang, H. Cheng, Y. Chen, Q. Chen, X. Lin, J. Qiu, Optimisation of reference gait trajectory of a lower limb exoskeleton, *Int. J. Soc. Robot.* 8 (2) (2016) 223–235.
- [14] F.M. Silva, J.A.T. Machado, Kinematic aspects of robotic biped locomotion systems, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 1997. IROS '97, 1997, pp. 266–272 vol.1.
- [15] E.T. Esfahani, M.H. Elahinia, Stable walking pattern for an SMA-actuated biped, *IEEE ASME Trans. Mechatron.* 12 (5) (2007) 534–541.
- [16] T. Kagawa, H. Ishikawa, T. Kato, C. Sung, Y. Uno, Optimization-based motion planning in joint space for walking assistance with wearable robot, *IEEE Trans. Robot.* 31 (2) (2015) 415–424.
- [17] D.E. Kirk, *Optimal Control Theory: An Introduction*, Courier Corporation, 2012.
- [18] Waijung Blockset. [Online]. Available: <http://waijung.aimagin.com/> [Accessed: 05-Jul-2018].
- [19] M. Agheli, S.S. Nestinger, Force-based stability margin for multi-legged robots, *Robot. Auton. Syst.* 83 (2016) 138–149.



**Jafar Kazemi** received his B.Sc. in electrical engineering from Tabriz University, Tabriz, Iran, in 2012, and his M.Sc. in electrical engineering from Tarbiat Modares University, Tehran, Iran, in 2014. He is currently a Ph.D. student in the Electrical and Computer Engineering Department at Tarbiat Modares University, Tehran, Iran. His current research interest focuses on the lower limb exoskeletons.



**Sadjaad Ozgoli** is an associate professor working with the electrical engineering school at Tarbiat Modares University. He works on control systems with application to Mechatronics systems and Biological systems. He received his B.Sc. and M. Eng degrees in electrical engineering from Sharif University of Technology, Tehran, Iran, respectively in 1997 and 1999. He received his Ph.D. from K.N. Toosi University of Technology, Tehran, Iran in electrical engineering with minors in mechanical engineering in 2005. He has published more than 40 papers in international journals and conference proceedings.