

# ZMP Theory-based Gait Planning and Model-Free Trajectory Tracking Control of Lower Limb Carrying Exoskeleton System

Haoping WANG, Yang TIAN\*, Shuaishuai HAN, Xikun WANG

School of Automation, Sino-French International Joint Laboratory of Automatic Control and Signal Processing (LaFCAS), Nanjing University of Science & Technology  
Nanjing, 210094, China

\*Corresponding author: tianyang@njust.edu.cn

**Abstract:** A virtual prototype design and corresponding mathematical model analysis is presented. To obtain a stable and smooth synthesized gait, the zero-moment-point (ZMP) theory and three spline interpolation method have been combined and applied. A novel data-driven model free control strategy, called Time-Delay Estimation based intelligent PID (TDE-iPID) combined with Equivalent Sliding Mode Control (TDE-iPIDESMC), is developed for the trajectory tracking of the planned gait. To demonstrate the effectiveness of the proposed planning gait and controllers, the exoskeleton system is implemented on the co-simulation platform composing of Adams and MATLAB/Simulink. The corresponding co-simulation results show better performances than the compared classical PID and TDE-iPID methods.

**Keywords:** lower limb carrying exoskeleton, gait planning, ZMP, TDE-iPIDESMC, co-simulation.

## 1. Introduction

As a wearable device, exoskeleton can provide physical support, motion assistance, strength enhancement and other functions. In the past decades, works on lower limb carrying exoskeleton can mainly be categorized into two types: (1) walking strength enhancement or load carrying, and (2) rehabilitation for gait disorder, spinal cord injury (SCI) or aged people. Because of the aging and human force enhancement needs, most of the existing exoskeletons fall into the first type, such as BLEEX and XOS, etc [14].

The application to rehabilitation medicines has received increasing attention in recent years. Researchers focus on developing walking aid systems or patient carrier system. Patients with gait disorder or SCI and elderly people who are unable to walk without assistance may lose muscular strength in their legs and become bedridden.

People are still hoping that exoskeleton attached to patients' legs can provide gait training, enhance their muscular strength or carry them to walk as normal individuals. Up to now, many different pro-types have been under research, such as LOKOMAT [1, 8], LOPES [19], HAL [12, 13], ReWalk [9], eLEGs and PRMI [3] which are all representative models.

The lower limb exoskeleton designed in our research focuses on the rehabilitation with the aim of walking aid and patient carrying. Stable planned gait and stability compensation of attitude are important for our exoskeleton in the situation of carrying disabled patient and walking normally in daily life. Although ZMP theory has been widely used in gait planning, most of existing achievements are based on limited knowledge of dynamics by using inverted pendulum model or five sticks model. To obtain planned gait with better performance and achieve more anthropomorphic walking, the exoskeleton is equipped with ten DOFs.

Conventional methods, such as computed torque control, impedance control and position control are usually applied to manipulators [2, 7]. More advanced control methods have been proposed recently. Based on the dynamic model with position and velocity errors, the nonlinear computed torque control can be applied [17]. In [18], the force-position control relies on the inputted force signal and a non-linear sliding mode control with exponential reaching law. Adaptive backstepping control based on fuzzy approximation makes it possible for exoskeleton to track the desired trajectory in presence of parametric uncertainties, unmodeled dynamics, and external disturbances [15, 22]. However,

most of the existing control strategies depend on the precise dynamical model. It is hard to get the precise mathematical model and usually a large deviation exists. For the exoskeleton in this research, each leg has three different dynamic models switching during walking period. The non-linear and strongly coupled switching dynamic model makes it hard to use model-based control strategies. The controllers designed here will be model-independent and robust to the changes of dynamic model, unmodeled dynamics and external disturbances. Developed by Fliess, model free control (MFC) theory has been applied to many applications [6, 21, 23, 24, 26, 27]. An algebraic technique is applied to estimate the unknown part in its corresponding ‘*ultra-local model*’. MFC is applied to an upper limb exoskeleton and the linearity technique is used for controller design in [21], without mentioning the data measurement technique. Such control idea is implemented to a twin rotor aerodynamic system in [24] as well and the unknown part is obtained by filtering method. Combined with backstepping technique and Linear Quadratic Regulator (LQR) theory, MFC is applied to a quadrotor UAV [27]. To avoid the computing burden in algebraic technique and achieve the effective estimation of unknown part in such a complex system, a novel data-driven MFC method based on time-delay estimation (TDE) is proposed in name of TDE intelligent PID (TDE-iPID). Besides, to compensate the estimation error and improve the robustness of the proposed controller, the equivalent sliding mode control (ESMC) is combined to TDE-iPID to form the final TDE-iPIDESMC.

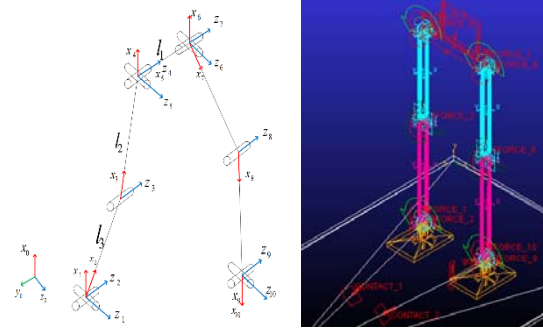
The remainder of this paper is organized as follows. Section II presents the design and model description of the virtual prototype. Section III gives the gait planning based on ZMP theory. In section IV, the model-free control methods of Time-Delay based intelligent PID (TDE-iPID) combined with Equivalent Sliding Mode Control (TDE-iPIDESMC) is proposed to realize the trajectory tracking. Co-simulation results are shown in section V and section VI presents some conclusion remarks.

## 2. Virtual Prototype Design and Mathematical Modeling

### Virtual prototype design

To provide rehabilitation therapy for the patients, the exoskeleton should possess a bio-

mimetic structure. The structure of the proposed exoskeleton has ten DOFs: two for ankle (varus/eversion, dorsiflexion/plantarflexion), one for knee (flexion/extension), and two for hip (flexion/extension, lateral/medial rotation). Fig.1.a) illustrates the DOF configuration and the virtual prototype is finally established as Figure 1.b) in ADAMS.



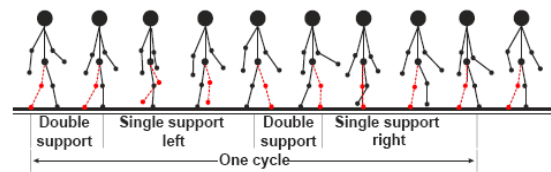
a) 10 DOFs configuration b) Virtual prototype

**Figure 1.** DOFs configuration and virtual prototype

### Mathematical model

Forward walking period can be divided into four phases [22] shown as in Figure 2. Dynamic model switch is used to describe different models in different phases. The commutations between the discrete phases are marked by collision and detachment events [16].

Collisions occur when shifting from the single to double support and detachment events occur when transitioning from double to single support.



**Figure 2.** Schemas of walking phase

### Kinematic model

According to the Denavit-Hartenberg (D-H) theory and coordinate systems built at each DOF, the D-H parameters can be obtained as Table 1. Between different coordinate systems, a  $4 \times 4$  matrix is introduced to build their relationship. The D-H translation matrix from coordinate system  $i$  to  $i-1$  is

$${}^{i-1}T_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_{i-1} \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -d_i \sin \alpha_{i-1} \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & d_i \cos \alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

And accordingly from coordinate system  $i$  to  $j$  is  ${}^jT_i = {}^jT_{j+1} \cdots {}^{i-2}T_{i-1} {}^{i-1}T_i$ ,  $i > j$ .

**Table 1.** D-H parameters of exoskeleton model

Number	$\theta_i$	$\alpha_{i-1}$	$a_{i-1}$	$d_i$
1	$\theta_1$	0	0	0
2	$\theta_2$	$90^\circ$	0	0
3	$\theta_3$	0	$l_3$	0
4	$\theta_4$	0	$l_2$	0
5	$\theta_5 - 90^\circ$	$-90^\circ$	0	0
6	$\theta_6 + 90^\circ$	0	$l_1$	0
7	$\theta_7 + 180^\circ$	$90^\circ$	0	0
8	$\theta_8$	0	$l_2$	0
9	$\theta_9$	0	$l_3$	0
10	$\theta_{10}$	$90^\circ$	0	0

The relationship between each joint can be established by the given D-H parameters, i.e., the forward kinematics. However, during the walking tasks, the foot position (terminal position) will be planned and the controller has to calculate the joint angles according to terminal position, i.e., the inverse kinematics. During forward walking process, the connection between two hips keeps parallel to the ground. Thus the lateral joint angles of ankle and hip can be considered equal in numerical value but with inverse direction, and the sum of ankle's forward joint angle, knee's joint angle, hip's forward joint angle between foot and ground in sagittal plane is equal to zero. In such a case, the following equation can be obtained

$$\begin{aligned} \theta_4 &= -\theta_2 - \theta_3 & \theta_9 &= -\theta_7 - \theta_8 \\ \theta_5 &= -\theta_1 & \theta_{10} &= -\theta_6 \end{aligned} \quad (2)$$

Using terminal position information  $(x, y, z)$ ,  $\theta_1, \theta_2, \theta_3$  ( $\theta_6, \theta_7, \theta_8$ ) can be further obtained:

$$\begin{aligned} q_1 &= \arctan(y/x) \\ q_2 &= \arctan 2(l_3 \sin q_3, l_2 + l_3 \cos q_3) \\ &\quad - \arctan 2[(x \cos q_1 + y \sin q_1) \sin q_3 \\ &\quad - z \cos q_3, (x \cos q_1 + y \sin q_1) \cos q_3 + z \sin q_3] \\ q_3 &= -\arccos \frac{(x \cos q_1 + y \sin q_1)^2 + z^2 - l_2^2 - l_3^2}{2l_2 l_3} \end{aligned} \quad (3)$$

The inverse kinematic model accomplish the solution from terminal space to joint space.

### Dynamic model

The dynamic model which describes the relationship between the torques and joint trajectories. Usually, the dynamical formulation of exoskeleton system can be expressed as

$$\tau = M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) \quad (4)$$

where  $M(\theta)$  is related inertia matrix term,  $C(\theta, \dot{\theta})$  denotes the Coriolis and Centrifugal matrix and  $G(\theta)$  represents the effects of gravity.

For swing leg, the dynamic model can be formulated as differential equations for controller application. Then equation (4) can be rewritten in the following form of

$$\ddot{\theta} = M^{-1}(\theta)[\tau - C(\theta, \dot{\theta}) - G(\theta)] \quad (5)$$

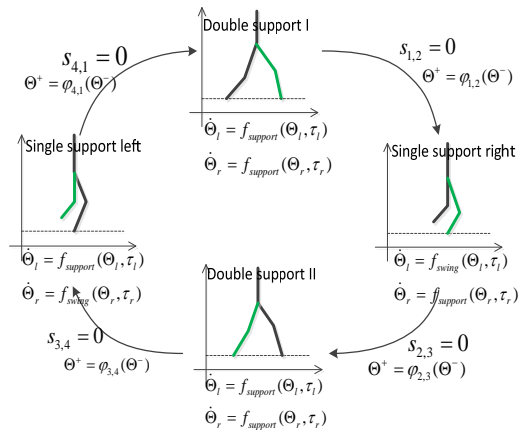
Defining the state vector  $\Theta = [\theta, \dot{\theta}]^T$ , the ordinary differential equations (ODEs) of swing leg can be obtained and denoted as follow

$$\dot{\Theta} = [\dot{\theta}, \ddot{\theta}]^T = f_{swing}(\Theta, \tau) \quad (6a)$$

For supporting leg, by considering the swing leg effects, its corresponding dynamics can be also established with similar equation as

$$\dot{\Theta} = f_{support}(\Theta, \tau) \quad (6b)$$

The state vector is reinitialized using the terminal state of prior phase after transition as illustrated in Figure3.



**Figure 3.** Transition between phases of exoskeleton

## 3. Gait Planning

There are three main ways to obtain exoskeleton's gait so far, gait planning based on Human Motion Capture Data (HMCD), ZMP theory and energy consumption. For the lower limb carrying exoskeleton designed here, walking stability will be the most vital issue, so ZMP theory is used here to ensure the stable planned gait. By importing gait data into

ADAMS, the stability of planned gait can be verified through dynamic simulation.

### ZMP Theory

The ZMP is originally defined for ground contacts of legs by Vukobratovic as the point in the ground plane about which the total moments due to ground contacts becomes zero in the plane [20]. If the ZMP stays in the convex hull of the all contacting points between the feet and the ground, the humanoid is possible to keep stable walking. ZMP theory can be applied to exoskeleton research from two technical aspects. One is generating dynamically consistent gait off-line based on ZMP, and the other is designing real-time stability compensation control. The referred ZMP can be calculated as follows [10]:

$$X_{zmp} = \frac{\sum_{i=1}^n m_i (\ddot{z}_i + g) x_i - \sum_{i=1}^n m_i \ddot{x}_i z_i}{\sum_{i=1}^n m_i (\ddot{z}_i + g)} \quad (7)$$

$$Y_{zmp} = \frac{\sum_{i=1}^n m_i (\ddot{z}_i + g) y_i - \sum_{i=1}^n m_i \ddot{y}_i z_i}{\sum_{i=1}^n m_i (\ddot{z}_i + g)} \quad (8)$$

with  $m_i$  represents mass of each part of lower limb,  $x_i, y_i, z_i$  are the mass center coordinates of each part.

### Parametric Gait Planning

The gait planning consists of the trajectory planning for each component parts during walking process [4, 11]. The trajectories of the hip joint (or upper body) and the ankle joint are planned firstly, and then the trajectories of the other components can be calculated through inverse kinematics.

During double support phase, both feet contact with ground. That starts with the front foot heel landing the ground, and ends with the rear toe off. During single support phase, supporting leg keeps still and the other leg swings forward. Since the latter two phases are similar to the front two phases, gait planning can be obtained from the front two phases.

For the sake of simplicity, various studies assume that the double support phases are transient. At that moment, the hip has to move quickly to maintain the stability of walking. However, longer period of double support will slow the walking speed. To get a suitable speed, the time of double support accounts for 10%-20% of the entire walking period  $2T_c$ .

where  $T_c$  represents half of the walking period, then step  $k+1$  starts at  $kT_c$  and ends at  $(k+1)T_c$ .

To simplify the gait planning, the Y coordinate of ankles is fixed ( $-l_1/2$  for left ankle and  $l_1/2$  for right ankle). Left foot heel leaves the ground at  $kT_c$ , and lands the ground at  $(k+1)T_c$ . The left leg's trajectory is planned firstly, and the right leg's trajectory has a time delay of  $T_c$ .  $(x_a, -l_1/2, z_a)$  is the coordinate of left ankle, while  $(x_h, y_h, z_h)$  is the coordinate of upper body. From the view of lowest energy consumption, it's expected that the swing leg lifts as low as possible. However, on the rough ground or in an environment with obstacles, the swing leg lifting a certain height is inevitable.

The left leg's gait planning is illustrated in Figure 4 which shows the begin and the end of double support phase, and the swing foot reaching the highest point, and the end of single support phase.

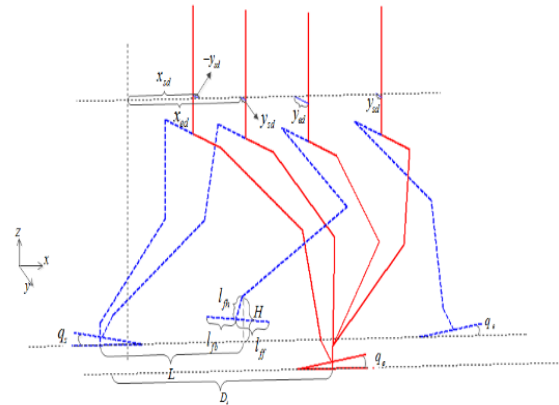


Figure 4. Gait planning of left leg (in blue)

Where  $T_d$  represents the time of double support phase,  $D_s$  represents the length of one step,  $q_s$  represents the angle when left foot leaves ground,  $q_e$  represents the angle when left heel lands ground, following constraint conditions can be summarized.

$$x_a(t) = \begin{cases} kD_s & t = kT_c \\ kD_s + l_{ff}(1 - \cos q_s) + l_{fh} \sin q_s & t = kT_c + T_d \\ kD_s + L & t = kT_c + T_o \\ (k+1)D_s - l_{fb}(1 - \cos q_e) - l_{fh} \sin q_e & t = kT_c + T_c \\ (k+1)D_s & t = kT_c + T_c + T_d \end{cases} \quad (9)$$

$$z_a(t) = \begin{cases} l_{fh} & t = kT_c \\ l_{fh} \cos q_s + l_{ff} \sin q_s & t = kT_c + T_d \\ H & t = kT_c + T_o \\ l_{fh} \cos q_e + l_{fb} \sin q_e & t = kT_c + T_c \\ l_{fh} & t = kT_c + T_c + T_d \end{cases} \quad (10)$$

Since the entire left foot plate has contact with ground at moments of  $kT_c$  and  $(k+1)T_c + T_d$ , then the following derivative constraints can be obtained after derivation:

$$\begin{cases} \dot{x}_a(kT_c) = 0 \\ \dot{x}_a((k+1)T_c + T_d) = 0 \end{cases} \quad \begin{cases} \dot{z}_a(kT_c) = 0 \\ \dot{z}_a((k+1)T_c + T_d) = 0 \end{cases} \quad (11)$$

In order to generate the smooth trajectory, the acceleration  $\ddot{x}_a(t)$  and  $\ddot{z}_a(t)$  must be continuous at any time. To avoid the complex calculation and possible oscillation in polynomial interpolation method, three spline interpolation method is adopted. During the hip trajectory planning,  $\theta_h(t)$  can be assigned to be  $90^\circ$  on the horizontal ground. It's hard for the movement of hip joint along the Z axis to influence the position of ZMP. For the sake of reducing impact between the foot plate and the ground and saving energy at the same time,  $z_h(t)$  is assigned to be a constant as well. Then  $x_h(t)$  and  $y_h(t) = (y_{hl}(t) + y_{hr}(t)) / 2$  need to be planned as smooth and stable trajectories. As illustrated in Figure 4, the following constraint conditions can be summarized:

$$x_{hi}(t) = \begin{cases} k/2 D_s + x_{sd} & t = kT_c \\ k/2 D_s + x_{ed} & t = kT_c + T_d \\ (k+1)/2 D_s + x_{sd} & t = kT_c + T_c \end{cases} \quad (12)$$

$$y_{hi}(t) = \begin{cases} -y_{sd} & t = kT_c \\ y_{sd} & t = kT_c + T_d \\ y_{ed} & t = kT_c + T_o \\ y_{sd} & t = kT_c + T_c \end{cases} \quad (13)$$

For a smooth trajectory, the following constraint conditions have to be satisfied:

$$\begin{cases} \dot{x}_{hi}(kT_c) = \dot{x}_{hi}((k+1)T_c) \\ \dot{y}_{hi}(kT_c) = \dot{y}_{hi}((k+1)T_c) \end{cases} \quad \begin{cases} \dot{y}_{hi}((k-1)T_c + T_o) = 0 \\ \dot{y}_{hi}(kT_c + T_o) = 0 \end{cases} \quad (14)$$

**Parameter optimization using genetic algorithm**

Although three interpolation method ensures the smoothness of the trajectories, optimization procedure is still needed to guarantee enough

stability margin. As an efficient optimization technique, the genetic algorithm is adopted here.

For single supporting phase, the stability margin can be represented as

$$f(d_x, d_y) = ad_x^2 + bd_y^2 \quad (15)$$

while  $d_x$  and  $d_y$  represents the distance between supporting phase centre and ZMP along X and Y axis (see Figure5). Since stability margin along X axis is larger than that along Y axis, weights  $a$  and  $b$  are used in the function, which satisfies

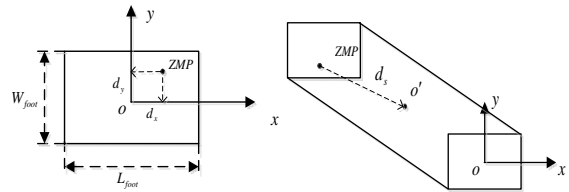
$$a/b = W_{foot} / L_{foot} \quad \text{and} \quad a + b = 1 \quad (16)$$

Using sampled data, the final optimization objective function during single supporting can be formulated as

$$J_s = a \sum_{i=1}^n d_x^2 + b \sum_{i=1}^n d_y^2 \quad (17)$$

For double supporting phase, the stability margin can be represented by the distance between supporting phase center and ZMP (Figure 5), then the objective function can be formulated as

$$J_d = \sum_{i=1}^m d_d^2 \quad (18)$$



**Figure 5.** Computation of stability margin

Then the total objective function becomes  $J = J_s + J_d$ , which will be the fitness function of genetic algorithm in form of  $f = C_{\max} - J$ . With the design of selection, the crossover and mutation operator, the optimized gait parameters can be obtained.

As a dynamic analysis software, ADAMS can generate and output the dynamic information of the virtual prototype during the gait tracking simulation experiment. So the stability of planned gait can be verified through this. But successful simulation needs start gait planning, which can be obtained similarly. Furthermore, considering the rigid sole of virtual prototype, the gait planned in section 3.2 needs to be

modified by keeping  $q_s$  and  $q_e$  being zero. The modified ankle's constraint conditions are

$$x_d(t) = \begin{cases} kD_s & t=kT_c \\ kD_s & t=kT_c+T_d \\ kD_s+L & t=kT_c+T_o \\ (k+1)D_s & t=kT_c+T_c \\ (k+1)D_s & t=kT_c+T_c+T_d \end{cases} \quad z_d(t) = \begin{cases} l_{fh} & t=kT_c \\ l_{fh} & t=kT_c+T_d \\ H & t=kT_c+T_o \\ l_{fh} & t=kT_c+T_c \\ l_{fh} & t=kT_c+T_c+T_d \end{cases} \quad (19)$$

#### 4. Data-driven based model free TDE-iPIDESMC control

As analysed above, the dynamic model of each leg constantly switches between swing leg and supporting leg during walking. Input and output signals are the only information obtained from the controlled plant, which makes it's even hard to design effective controllers for the system. As introduced in this part, the control of ten joints can be decoupled by the proposed TDE-iPIDESMC which is an efficient combination of the 'ultra-local model' idea, TDE technique, iPID, and equivalent sliding mode control.

##### Data-driven model free control

For the considered lower limb carrying exoskeleton systems whose dynamic models are complex illustrated in (6a-6b) may present uncertainties, external disturbance and parameters changing, thus this section will focus on the development of data-driven model free controller design. For the sake of simplicity, a single input and single output system can be considered here [21]. The unknown 'complex' mathematical model can be replaced by an 'ultra-local model' as follows

$$y^{(v)} = F + \alpha u \quad (20)$$

$y^{(v)}$  is the output derivative with order  $v \geq 1$ , i.e., the position information of each joint in this research. Their referred integer  $v$  is selected by the practitioner and generally its value is 1 or 2.  $u$  is the joint torque implemented to exoskeleton.  $\alpha$  is a non-physical constant parameter.  $F$  is continuously updated which constitutes the dynamics of poorly known plant and possible external disturbances.

Close the loop via the intelligent PID controller (iPID) which is designed as follow:

$$u = -\frac{F - y_d^{(v)} - (K_p e + K_i \int e + K_D \dot{e})}{\alpha} \quad (21)$$

where  $y_d^{(v)}$  represents the derivative of order  $v$  of the desired trajectory of each joint and  $e = y_d - y$ .

Combining equation (20) and (21), the following equation can be obtained

$$(y_d^{(v)} - y^{(v)}) + K_i \int e + K_p e + K_D \dot{e} = 0 \quad (22)$$

From equation (22), it's obvious that the unknown parts and uncertainties of the plant  $F$  is eliminated. The tracking error will approach zero by simple pole-zero assignment.

##### Time-delay estimation based TDE-iPID

Obviously the online estimation of unknown part  $F$  is the crucial issue for the MFC. The iterative method proposed in [5] is based on numerical differentiation technique which takes high computational cost. In this research, the online estimation is realized using a kind of data-driven technique with time-delay, which means the former input and output signals are used to calculate  $F$  in the form of

$$F(t) \approx \hat{F}(t) = F(t - \varepsilon) = y^{(v)}(t - \varepsilon) - \alpha u(t - \varepsilon) \quad (23)$$

where  $\varepsilon$  is a small time interval. Due to the dynamic analysis, the rates of change of the accelerations at all joints are bounded. With the continuity of  $F$ , equation (23) can be satisfied if  $\varepsilon$  is small enough. Decreasing  $\varepsilon$  will improve the estimation precision but increase the computing burden. Besides, the improvement will not be obvious when reduce  $\varepsilon$  to a particular value.

Thus, the data-driven model free Time-Delay Estimation based iPID (TDE-iPID) control can be defined as follow

$$u(t) = -\frac{y^{(v)}(t - \varepsilon) - \alpha u(t - \varepsilon) - y_d^{(v)}(t) - (K_p e(t) + K_i \int e(t) + K_D \dot{e}(t))}{\alpha} \quad (24)$$

During co-simulation experiments, the virtual prototype shows better performance when  $v = 2$ .

##### TDE-iPID based equivalent sliding mode control

Combining equation (20), (23) and (24), the estimation error can be obtained as

$$e_{est} = F - \hat{F} = -e^{(v)} - (K_p e + K_i \int e + K_D \dot{e}) = -e^{(v)} - u_{PID} \quad (25)$$

As mentioned that the performance of the proposed TDE-iPID control depends on the estimation precision. The tracking error may not converge to zero because of the existence of  $e_{est}$ , but can be limited in a relatively small

bound i.e.,  $|e_{est}| \leq E$  with  $E$  a constant positive value.

As shown in Figure6, the Equivalent Sliding Mode Control (ESMC)  $u_{smc}$  is added to TDE-iPID control in (24) to compensate the estimation error  $e_{est}$ , and the obtained new iPID-ESMC control is defined as follow:

$$u(t) = -\frac{\hat{F} - y^{(v)}_d(t) - (K_p e(t) + K_I \int e(t) + K_D \dot{e}(t))}{\alpha} + u_{smc} \quad (26)$$

From equation (24), the following closed loop equation can be obtained as

$$e_{est} = F - \hat{F} = -e^{(v)} - (K_p e + K_I \int e + K_D \dot{e}) - \alpha u_{smc} \quad (27)$$

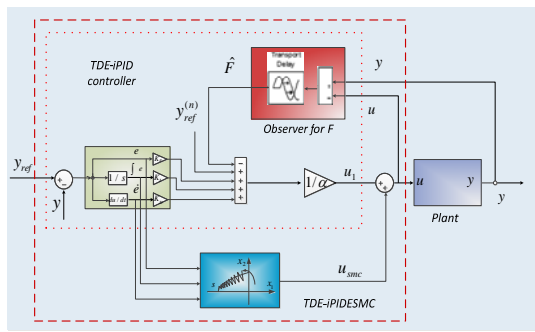


Figure 6. Structure of TDE-iPIDESMC control

For  $v = 2$ , equation (25) can be rewritten as

$$\begin{cases} x_1 = e \\ x_2 = \dot{e} \end{cases} \text{ and } \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -K_p x_1 - K_I \int x_1 - K_D x_2 - \alpha u_{smc} - e_{est} \end{cases} \quad (28)$$

with the linear sliding mode surface

$$s = c e + \dot{e} \quad (29)$$

where  $c > 0$ . After reaching the sliding mode surface, the tracking error  $e$  will asymptotically approach zero. Its derivative can be calculates as follow

$$\dot{s} = c \dot{e} + \ddot{e} = c \dot{e} - K_p e - K_I \int e - K_D \dot{e} - \alpha u_{smc} - e_{est} \quad (30)$$

As usual, the Lyapunov function is chosen as

$$V = \frac{1}{2} s^2, \text{ then under the condition } \dot{V} = s \dot{s} < 0,$$

the ESMC which consists of equivalent control  $u_{eq}$  and switching control  $u_{sw}$  is defined as

$u_{smc} = u_{eq} + u_{sw}$ , in which the equivalent control can be obtained by setting  $\dot{s} = 0$  as:

$$u_{eq} = \frac{c \dot{e} - K_p e - K_I \int e - K_D \dot{e}}{\alpha} \quad (31)$$

and the switching control ensures the reachability to the sliding mode surface, which means:

$$s \dot{s} \leq -\eta |s| \quad \text{with} \quad \eta > 0 \quad (32)$$

Then the switching control can be chosen as

$$u_{sw} = \frac{1}{\alpha} K \text{sgn}(s) \quad \text{with} \quad K = E + \eta \quad (33)$$

Finally the compensation control becomes

$$u_{smc} = u_{eq} + u_{sw} = \frac{c \dot{e} - K_p e - K_I \int e - K_D \dot{e}}{\alpha} + \frac{E + \eta}{\alpha} \text{sgn}(s) \quad (34)$$

Substituting (34) to (30)

$$\begin{aligned} \dot{s} &= c \dot{e} - K_p e - K_I \int e - K_D \dot{e} \\ &\quad - [c \dot{e} - K_p e - K_I \int e - K_D \dot{e} + (E + \eta) \text{sgn}(s)] - e_{est} \\ &= -(E + \eta) \text{sgn}(s) - e_{est} \end{aligned} \quad (35)$$

Consequently, with  $|e_{est}| \leq E$ , we have

$$\begin{aligned} \dot{V} &= s \dot{s} = s [-(E + \eta) \text{sgn}(s)] - s \cdot e_{est} \\ &= -|s| (E + \text{sgn}(s) \cdot e_{est}) - \eta |s| \\ &\leq -\eta |s| \\ &\leq 0 \end{aligned} \quad (36)$$

where  $\dot{V} = 0$  only happens when  $s = 0$ . Because of the negative definition of  $V$ , the control law will drive the tracking error  $e$  to the sliding mode surface (29) and then realize the zero error tracking.

Finally, the TDE-iPIDESMC is summarized as

$$u(t) = -\frac{\hat{F} - \ddot{y}_d(t)}{\alpha} + \frac{c \dot{e}}{\alpha} + \frac{E + \eta}{\alpha} \text{sgn}(s) \quad (38)$$

Furthermore, the sign function will be replaced by suitable saturation function to reduce the chattering.

$$\text{sat}(s) = \begin{cases} 1 & s > \Delta \\ ks & |s| \leq \Delta \\ -1 & s < -\Delta \end{cases} \quad k = 1/\Delta \quad (39)$$

where  $\Delta$  denotes boundary set.

## 5. Co-simulation results

In order to validate the proposed TDE-iPIDESMC method, the virtual prototype established in ADAMS was exported to MATLAB/Simulink as the controlled plant.

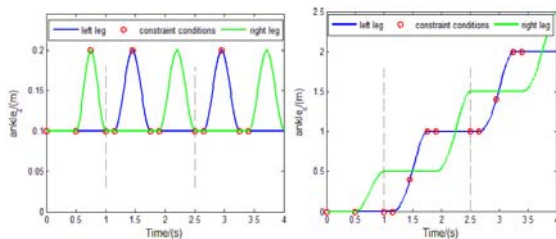
The parameters used in the virtual prototype are based on the average level of anthropometric parameters and the corresponding human body weight is directly added to exoskeleton. During the co-simulation, the output signals (angle, velocity, acceleration) are generated by ADAMS and control torque is directly applied to each joint.

**Gait planning with start gait**

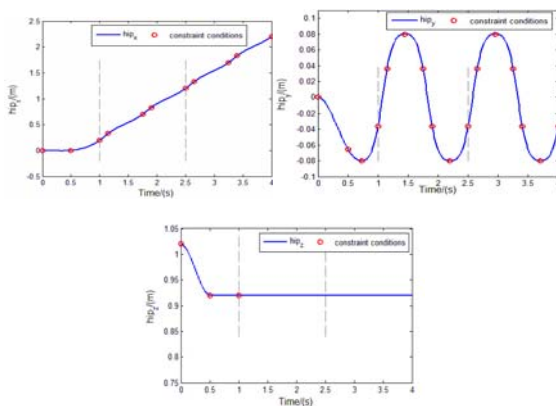
With the application of the proposed ZMP and genetic algorithm based optimal gait planning method, the  $x_{sd}$   $x_{ed}$   $y_{sd}$  and  $y_{ed}$  ensuring  $X_{zmp}$  and  $Y_{zmp}$  within support phase are  $x_{sd}=0.2m, x_{ed}=0.33m, y_{sd}=0.036m, y_{ed}=0.08m$ .

As mentioned, the start gait needs to be planned as well to enable the exoskeleton to walk from its initial standing state.

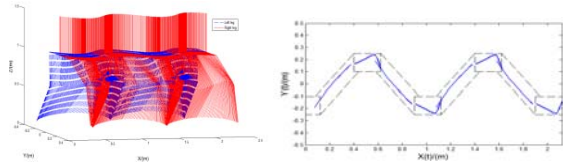
With the proposed ZMP theory based gait planning algorithm in Section 3, its corresponding planned normal walking trajectories (two periods) with start gait are generated and shown in Figure 7-8 respectively. And its applied ZMP trajectory and stick figure (in two periods) shown in Figure 9.



**Figure 7.** Gait of ankle along X axis and Z axis



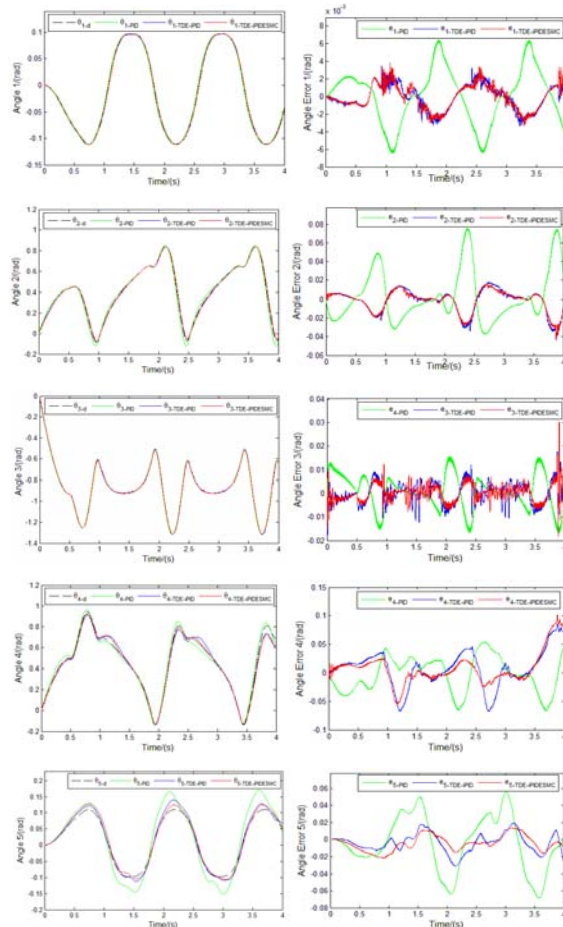
**Figure 8.** Gait of hip along X, Y and Z axis



**Figure 9.** a). Stick figure (left) b). ZMP trajectory and convex hull (right)

**TDE-iPIDESMC trajectory tracking results**

Through inverse kinematic, the trajectories of terminal can be transformed into joint space. With the proposed control strategy, the trajectory tracking can be realized and the corresponding trajectory tracking results of each DOF are implemented through traditional PID, TDE-iPID and TDE-iPIDESMC control for validation. Only the right leg's tracking results are illustrated here for the exoskeleton's symmetry. Figure 10 shows the responses of right leg (joint 1 to 5) with their corresponding joint trajectory tracking errors. TDE-iPIDESMC demonstrates better tracking performances than the compared PID and TDE-iPID and their corresponding control torques are illustrated in Figure 11 which show that the proposed TDE-iPIDESMC has moderate amplitudes.



**Figure 10.** The tracking results (left) and corresponding error (right)



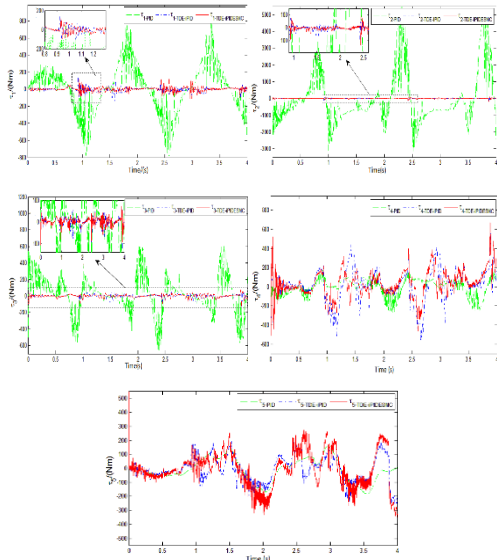


Figure 11. Control torque of each joint

### Animations of stable walking

In co-simulation environment, the walking performance can be recorded by animation films. The screen-shot animations of corresponding walking on horizontal ground and upstairs are depicted respectively in Figure 12. The simulated animations further demonstrate the effectiveness of the gait planning.

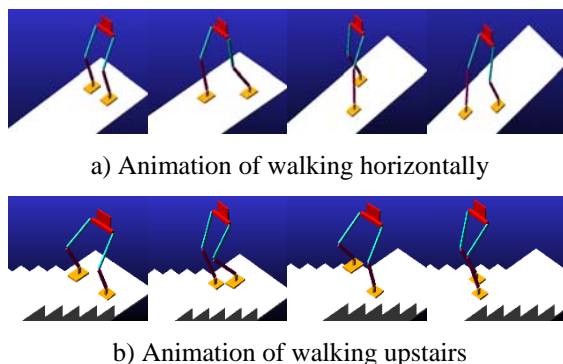


Figure 12. Animations of different tasks

## 6. Conclusions

This paper accomplishes firstly the lower limb exoskeleton design, model analysis, and ZMP theory based Gait planning, then considering different phases during forward walking which is treated as switching system, a Data-driven model free TDE-IPIDEMC trajectory tracking control is developed. Finally, with the developed model free control, the trajectory tracking of the planned gaits is concluded with the insurance of the stability. Moreover, the robustness and effectiveness of the proposed

TDE-IPIDEMC is demonstrated with co-simulation results compared to PID and TDE-iPID methods.

## Acknowledgements

This work was partially supported by International Science & Technology Cooperation Program of China (2015DFA01710), the National Natural Science Foundation of China (61304077), by the Chinese Ministry of Education Project of Humanities and Social Sciences (13YJCZH171), and by the 11th Jiangsu Province Six talent peaks of high level talents (2014\_ZBZZ\_005).

## REFERENCES

1. Agrawal, S. K., Banala, S. K. & Fattah, A. (2007). Assessment of Motion of a Swing Leg and Gait Rehabilitation With a Gravity Balancing Exoskeleton. *IEEE Transactions on Neural Systems & Rehabilitation Engineering*, 15(3), 410-20.
2. Janne, K. & Jouni, M. (2017). Stability-Guaranteed Impedance Control of Hydraulic Robotic Manipulators. *IEEE/ASME TRANSACTIONS ON MECHATRONICS*, 22(2), 601-612.
3. Duong M. K., Hong C. & Tran H. T. (2015). Minimizing human-exoskeleton interaction force using compensation for dynamic uncertainty error with adaptive RBF network. *Journal of Intelligent & Robotic Systems*, 1-21.
4. Feng S. & Sun Z. (2008). Biped robot walking using three-mass linear inverted pendulum model. *Intelligent Robotics and Applications*, 371-380.
5. Fliess, M. & Join, C. (2013). Model-free control. *International Journal of Control*, 86(12), 2228-2252.
6. Fliess, M. & Join, C. (2014). Stability margins and model-free control: A first look. *European Control Conference IEEE*, 454-459.
7. Gao A. Z., Ryan J. M., Liu H., Lulian L. & Mehran A. (2017). Mechanical Model of Dexterous Continuum Manipulators with Compliant Joints and Tendon/External Force Interactions. *IEEE/ASME TRANSACTIONS ON MECHATRONICS*, 22(1), 465-475.

8. Hidler, J., Wisman, W. & Neckel, N. (2008). Kinematic trajectories while walking within the Lokomat robotic gait-orthosis. *Clinical Biomechanics*, 23(10), 1251-1259.
9. <http://rewalk.com/>
10. Huang Q, Yokoi K. & Kajita S. (2001). Planning walking patterns for a biped robot. *IEEE Transactions on Robotics & Automation*, 17(3), 280-289.
11. Mostafa, E., Aghil, Y. & Majid K. (2016) A Novel Three-Mass Inverted Pendulum Model for Real-Time Trajectory Generation of Biped Robots. *Proceedings of the 4th International Conference on Robotics and Mechatronics*, 404-409.
12. Kasaoka, K. & Sankai, Y. (2001). Predictive control estimating operators intention for stepping-up motion by exoskeleton type power assist system HAL. *IEEE International Conference on Intelligent Robots and Systems*, 1578-1583.
13. Kawamoto, H. & Sankai, Y. (2002). Comfortable power assist control method for walking aid by HAL-3. *IEEE International Conference on Systems, Man and Cybernetics*, 2002: 6.
14. Kazerooni H. (2011). Human augmentation and exoskeleton systems in berkeley. *International Journal of Humanoid Robotics*, 4(3), 575-605.
15. Li Z., Su C. Y. & Li G. (2014). Fuzzy Approximation - Based Adaptive Backstepping Control of an Exoskeleton for Human Upper Limbs. *IEEE Transactions on Fuzzy Systems*, 23(3), 1-1.
16. Malita, V., Misgeld, B. J. E. & Leonhardt, S. (2014). Design of an adaptive gait trajectory controller based on a hybrid two-legged robot model. *International Conference on Methods and Models in Automation and Robotics*, 681-686.
17. Rahman, M. H., Saad, M. & Kenné, J. P. (2009). Modeling and control of a 7DOF exoskeleton robot for arm movements. *IEEE International Conference on Robotics and Biomimetics*, 245-250.
18. Rahman, M. H., Ochoaluna, C. & Rahman, M. J. (2014). Force-position control of a robotic exoskeleton to provide upper extremity movement assistance. *International Journal of Modelling Identification & Control*, 21(4), 390-400.
19. Vallery, H., Ekkelenkamp, R. & Buss M. (2007). Complementary limb motion estimation based on interjoint coordination: experimental evaluation. *10th IEEE International Conference on Rehabilitation Robotics*, 798-803.
20. Vukobratovic, M. & Juricic, D. (1969). Contribution to the Synthesis of Biped Gait. *IEEE Transactions on Biomedical Engineering*, BME-16(1):1-6.
21. Wang X. F., Li X, Wang J.H Fang X. K. & Zhu X. F. (2016). Data-driven model-free adaptive sliding mode control for the multi degree-of-freedom robotic exoskeleton. *Information Sciences*, 327(C), 246-257.
22. Wang Z. F., Peyrodie L., Cao H. & Wang H P. (2016). Slow walking model for children with multiple disabilities via an application of humanoid robot. *Mechanical Systems & Signal Processing*, 68, 608-619.
23. Younes, Y. A., Drak, A., Noura, H., Rabhi, A. & Hajjaji A. E. (2014). Model-free control of a quadrotor vehicle. *International Conference on Unmanned Aircraft Systems. IEEE*, 1126-1131.
24. Radu-Emil, P., Mircea-Bogdan, R., Raul-Cristian, R. & Emil, M. P. (2017). Model-free sliding mode control of nonlinear systems: Algorithms and experiments. *Information Sciences*, 381, 176-192.
25. Hou Z. S. & Zhu Y. M. (2013). Controller-Dynamic-Linearization-Based Model Free Adaptive Control for Discrete-Time Nonlinear Systems. *IEEE Transactions on Industrial Informatics*, 9(4): 2301-2309.
26. Ouassim, B., Michel, F., Cedric, J., Judy, D. & Seddik M. (2016). Model-free immune therapy: A control approach to acute inflammation. *European Control Conference (ECC)*, 2102-2107.
27. Younes, A.Y., Ahmad, D., Hassan, N., Abdelhmid R. & Ahmed E. H. (2016). Robust Model-Free Control Applied to a Quadrotor UAV. *Journal of Intelligent and Robotic Systems*, 84(1-4).