Motion Control Algorithm for a Lower Limb Exoskeleton Based on Iterative LQR and ZMP method for trajectory generation

S. Jatsun[1], S. Savin[1] and A. Yatsun[1]

[1] *Department of Mechanics, Mechatronics and Robotics of Southwest State University, Kursk, Russia, e-mail: teormeh@inbox.ru.*

Abstract. In this paper a problem of controlling a lower limb exoskeleton during sit-to-stand motion (verticalization) in sagittal plane is studied. It is assumed that left and right sides of the exoskeleton are moving symmetrically. The main challenge in performing this motion is to maintain balance of the system. In this paper we use the zero-moment point (ZMP) methodology to produce desired trajectories for the generalized coordinates that would allow the system to remain vertically balanced. The limitations of this approach is that, it requires relatively accurate work of the feedback controller that ensures that the exoskeleton follows generated trajectories. In this work we use Iterative Linear Quadratic Regulator (ILQR) as a feedback controller in order to obtained the required accuracy. In the paper a way of trajectory generation that uses ZMP methodology is discussed, the results of the numerical simulation of the exoskeleton motion are presented and analyzed. A comparison between a natural human motion (for a human not wearing an exoskeleton) and the simulated motion of an exoskeleton using the proposed algorithm is presented.

Key words: lower limb exoskeleton, sit-to-stand motion, ZMP trajectory generation, iterative linear quadratic regulator, control system design.

1 Introduction

Exoskeleton development is an important research topic in robotics with a wide range of practical applications. Exoskeletons have been used for providing assistance to elderly and disabled people, rehabilitation, and for enhancing the performance of industrial workers and combatants. There are different challenges associated with those applications. The main problem is the necessity to guarantee the safety of the human [1, 2].

There are many aspects of safety problem for exoskeleton wearers, many of them are discussed in article [2]. One of the most crucial safety problems is maintaining vertical balance of the exoskeleton. A popular approach to solving this problem for walking robots is the use of zero-moment point (ZMP) control [3]. The methodol-

ogy of ZMP control as applied to bipedal robot and exoskeleton is discussed in papers [4-8]. Publication [9] presents the use of ZMP control to a lower limb exoskeleton during sit-to-stand motion. As stated in paper [5] some realizations of ZMP control can be criticized for lack of adaptive properties; and as mentioned in [10], the fact that ZMP method is often implemented using a simplified model of the system can lead to additional control errors. We note that solutions to both of these problems are proposed in the same papers. Paper [11] describes the implementation of optimization-based control that uses information about the ZMP position, which was used on the Atlas robot during the DARPA Robotics Challenge. While this approach is computationally intensive for an on-board system, it can provide a way to further improve the safety of lower-limb exoskeletons, especially in the case of assistive exoskeletons for paraplegic patients.

In this paper we focus on sit-to-stand motion. The execution of this motion puts high demands on the exoskeleton's control system, making the problems discussed above more challenging. For medical and assistive exoskeletons the task of standing up has additional importance, because its automation removes the need of human assistance for exoskeleton users [12-13]. Publications [12-17] discuss different aspects of sit-to-stand motion, including problems of control. Adaptive feedback controllers for exoskeletons performing sit-to-stand motion are studied in papers [18-19].

The goal of this paper is to study a control strategy uses ZMP trajectory generator and Iterative Linear Quadratic Regulator (ILQR).

2 Description of the exoskeleton

An exoskeleton consists of the following basic elements: 1 – a corset that fixes the exoskeleton to the patient's body by means of straps, 2 – thigh links installed on the corset by means of two two-axes hinges, one of which is equipped with an electric drive, 3- shin link fixed onto the hips by means of actuated rotary joint, 4- feet links attached to the shin by means of a three-axes hinge, two of which are equipped with rotary electric drives. All the electric drives have current sensors, which allow us to control the values of torques generated by these electric drives.

Fig. 1 General view of the exoskeleton (3D model) 1- exoskeleton torso, 2 – thigh link, 3 – shin link, 4 – foot link, 5-9 – connecting belts.

All the hinges of the exoskeleton are equipped with rotation angle sensors (absolute encoders), and each motor's rotor is additionally equipped with an incremental encoder. The feet contain pressure sensors that measure normal reactions. Information from the sensors is preprocessed using onboard microcontrollers and then the data is send to the onboard PC that realizes the control system.

3 Mathematical model of the exoskeleton

In this chapter, we consider a mathematical model of the lower limb exoskeleton performing sit-to-stand motion in the sagittal plane. We make an assumption that during the sit-to-stand motion both leg links, as well as all the thigh links move parallel to each other. We assume that the feet remain motionless for the whole duration of verticalization. This allows us to model the exoskeleton as a four link serial mechanism, as shown in fig. 2.

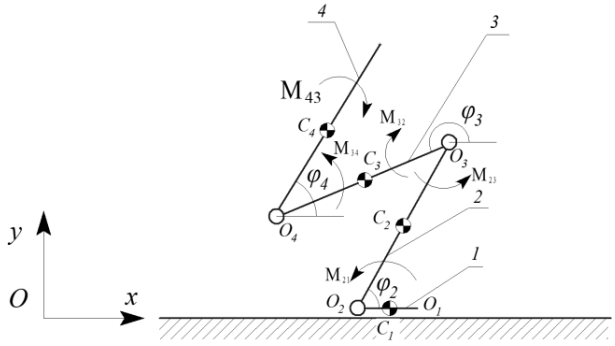

Fig. 2 Analytical diagram of the exoskeleton; 1-4 – first – fourth links.

On the diagram in fig. 2 points O_2 - O_4 are rotational joints that connect the links, points C_1 - C_4 are the centers of mass of the corresponding links, φ_2 - φ_4 are the angles that determine the orientation of the corresponding links. The *i*-th link has mass m_i, length l_i and inertia I_i. The mass of the whole mechanism is denoted as $m = \sum_{i=1}^{4} m_i$.

Using the introduced notations we can write the equation that describes the position of the center of mass of the mechanism:

$$\begin{cases} x_C = x_{O1} + \sum_{i=1}^{4} \left(K_i \cos(\varphi_i)\right) \\ y_C = y_{O1} + \sum_{i=1}^{4} \left(K_i \sin(\varphi_i)\right) \end{cases} \tag{1}$$

where x_C, y_C are the Cartesian coordinates of the robot's center of mass, x_{O1}, y_{O1} are the coordinates of point O_1 (the tip of the foot). Coefficients K_i are given as follows:

$$K_1 = l_1\left(M - 0.5m_1\right)/m\,, \tag{2}$$

$$K_2 = l_2\left(0.5m_2 + m_3 + m_4\right)/m\,, \tag{3}$$

$$K_3 = l_3\left(0.5m_3 + m_4\right)/m\,, \tag{4}$$

$$K_4 = 0.5l_4m_4/m\,. \tag{5}$$

To write the equations of motion we introduce a vector of generalized coordinates $\vec{q}$:

$$\vec{q} = [\varphi_2 \quad \varphi_3 \quad \varphi_4]^{\mathrm{T}}. \tag{6}$$

The dynamics of the robot then can be described by the following equations:

$$\mathbf{\Lambda}(\vec{q})\ddot{\vec{q}} + \vec{C}(\vec{q},\dot{\vec{q}}) + \vec{G}(\vec{q}) + \vec{\Phi}(\dot{\vec{q}}) = \mathbf{H}\vec{M}, \tag{7}$$

where $\dot{\vec{q}} = [\dot{\varphi}_2 \quad \dot{\varphi}_3 \quad \dot{\varphi}_4]^{\mathrm{T}}$ is a vector of generalized velocities, $\ddot{\vec{q}} = [\ddot{\varphi}_2 \quad \ddot{\varphi}_3 \quad \ddot{\varphi}_4]^{\mathrm{T}}$ is a vector of generalized accelerations, $\mathbf{\Lambda}(\vec{q})$ is a joint space inertia matrix, $\vec{C}(\vec{q},\dot{\vec{q}})$ is a vector of generalized forces of inertia, $\vec{G}(\vec{q})$ is a vector of generalized gravitational forces, $\vec{\Phi}(\dot{\vec{q}})$ - vector of generalized dissipative forces, $\vec{M} = [M_{21} \quad M_{32} \quad M_{43}]^{\mathrm{T}}$ is a vector whose components are the torques of the motors, $\mathbf{H}$ is a matrix connecting the torques of the motors with generalised forces they generate. The joint space inertia matrix $\mathbf{\Lambda}(\vec{q})$ can be found as the matrix of the quadratic form of kinetic energy $\mathbf{\Lambda}(\vec{q}) = \dfrac{\partial^2 T(\vec{q},\dot{\vec{q}})}{\partial \dot{\vec{q}}^2}$, where T is the kinetic energy of the system (see [20] for a complete treatment of these equations, and [21] for notation). The scalar form of the presented equations can be found in paper [22].

In the following chapters we will be considering the case when vector $\vec{M}$ is a function of control error.

For some of the formulations in the next chapter we need to write the dynamics in the state-space form. To do that we introduce a state vector $\vec{s}$:

$$\vec{s} = [\dot{\varphi}_2 \quad \dot{\varphi}_3 \quad \dot{\varphi}_4 \quad \varphi_2 \quad \varphi_3 \quad \varphi_4]^{\mathrm{T}}. \tag{8}$$

Using this notation we can write the following equations of motion in vector form:

$$\dot{\vec{s}} = f(\vec{s}) = \begin{bmatrix} \mathbf{\Lambda}^{-1}\left(\mathbf{H}\vec{M} - \vec{C} - \vec{G} - \vec{\Phi}\right) \\ s_1 \\ s_2 \\ s_3 \end{bmatrix} \tag{9}$$

The system of first-order ordinary differential equations (ODE) (9) can be used to simulate motion of the exoskeleton for given $\vec{M}$. in chapter 4.2 this system of equations will be used to derive Iterative Linear Quadratic Regulator (ILQR).

4 Control system design

In this chapter we discuss the proposed control system design. In the first part we the method of trajectory generation for the center of mass (CoM) during sit-to-stand motion. The method allows us to move zero moment point (from here onwards we denote it as point P) towards the center of the support polygon. In the second part we look at the augmented ZMP Control strategy.

4.1 ZMP trajectory generation

Establishing the desired trajectory of the CoM of the mechanism can be used as a step in formulating verticalization control task [18-19, 23]. The ZMP methodology presents tools to find the CoM's trajectory that allows the zero moment point P to stay within the support polygon. For the model described in the previous chapter the position of point P can be found from the following expression:

$$x_P = x_C - \left(mg + m\ddot{y}_C\right)^{-1}\left(m\ddot{x}_C y_C - \sum_{i=2}^{4} J_i\ddot{\varphi}_i\right) \tag{10}$$

where g is the gravitational acceleration.

Point P should stay within the support polygon in order to maintain the mechanism's balance, which means that the following inequalities should hold:

$$\left(x_{O1} - l_1\right) < x_P(t) < x_{O1}, \qquad t \in [0 \quad t_f] \tag{11}$$

where t_f is the duration of sit-to-stand motion.

We define the desired trajectory for point P in the following way:

$$x_P^*(t) = \begin{cases} \sum_{k=0}^{3} z_k \cdot t^k & if \quad t < t_1 \\ 0.5 \cdot l_1 & if \quad t \geq t_1 \end{cases} \tag{12}$$

where z_k are polynomial coefficients given by the following expressions:

$$z_0 = x_P(0),\ z_1 = 0,\ z_2 = \frac{3}{t_f^2}\left(x_{O1} - \frac{1}{2}l_1 - x_P(0)\right), \tag{13}$$

$$z_3 = \frac{2}{t_f^3}\left(x_P(0) - x_{O1} + \frac{1}{2}l_1\right), \tag{14}$$

where $x_P(0)$ is the initial position of point P.

Assuming that the desired motion of the CoM in the vertical direction $y_C^*(t)$ is given, for example as a polynomial function as was done in [18], we can use (10) and (12) to write an equation that can be solved for the CoM acceleration in the horizontal direction:

$$my_C^*\ddot{x}_C^* - m(g + \ddot{y}_C^*)(x_C^* - x_P) - \sum_{i=2}^{4} I_i\ddot{\varphi}_i^* = 0 \tag{15}$$

To find $\ddot{\varphi}_i^*$ we can use the following derivation:

$$\ddot{q}^* = \begin{bmatrix} \ddot{\varphi}_2^* \\ \ddot{\varphi}_3^* \\ \ddot{\varphi}_4^* \end{bmatrix} = \left(\frac{\partial \dot{r}_t}{\partial \dot{q}}\right)^{-1}\left(\ddot{r}_t^* - \frac{\partial \dot{r}_t}{\partial q}\dot{q}^*\right), \tag{16}$$

$$\dot{q}^* = \left(\frac{\partial r_t}{\partial q}\right)^{-1}\dot{r}_t^*, \tag{17}$$

where $r_t = [x_C \quad y_C \quad \varphi_4]^{\mathrm{T}}$, $\ddot{r}_t^* = [\ddot{x}_C^* \quad \ddot{y}_C^* \quad \ddot{\varphi}_4^*]^{\mathrm{T}}$ and $\dot{r}_t^* = [\dot{x}_C^* \quad \dot{y}_C^* \quad \dot{\varphi}_4^*]^{\mathrm{T}}$.

After substituting (16) into (15), we obtain a second order linear ODE that can be solved numerically or analytically to obtain the desired trajectory $x_C^*(t)$. In practice it might be beneficial to add a damping term to the equation to decrease the oscillations in the solution.

4.2 Iterative Linear Quadratic Regulator

Using the method of trajectory generation for the CoM described above it is possible to build a control system that drives the CoM along that trajectory. Fig. 3 shows a design of such a control system.

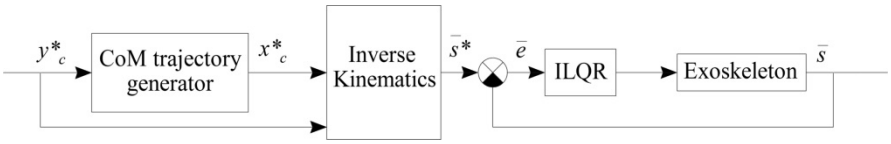

Fig. 3 ZMP control design.

The CoM Trajectory Generator module determines $x_C^*(t)$ and its derivatives by solving (15); the Inverse Kinematics module finds values of generalized coordinates that cause the CoM to move along the desired trajectory. The analytical solution of this problem can be found in [18]. The Robot module represents the dynamics of the mechanism as described by (7). The ZMP Finder module uses (10) to find the ZMP's current position. Vector $\vec{e}$ is the control error, and it is determined as $\vec{e} = \vec{s}^* - \vec{s}$.

The work of the ILQR module is described below. ILQR is an iterative control algorithm that requires local linearization of the system model (for detailed description see [23]). Here we consider the case when the new linearization is made on every iteration. The linearized model of the system has form:

$$\dot{\vec{s}} = \mathbf{A}\vec{s} + \mathbf{B}\vec{M} + \vec{b}\,, \tag{18}$$

where $\mathbf{A} = \partial\vec{f}/\partial\vec{s}$, $\mathbf{B} = \partial\vec{f}/\partial\vec{M}$ (both evaluated at the point $\vec{s} = \vec{s}^*(t)$), $\vec{b}$ is the vector that describes the difference between the actual model of the system and its linearized version. Vector $\vec{b}$ can be found using the following expression:

$$\vec{b} = \vec{f}(\vec{s}) - \mathbf{A}\vec{s} - \mathbf{B}\vec{M}\,, \tag{19}$$

Vector function $\vec{s}^*(t)$ denotes the desired position of the system and it has form:

$$\vec{s}^*(t) = [\dot{\vec{q}}^*(t) \quad \vec{q}^*(t)]^{\mathrm{T}}\,. \tag{20}$$

We define quadratic cost function J:

$$J = \int[(\vec{s} - \vec{s}^*)^{\mathrm{T}}\mathbf{Q}(\vec{s} - \vec{s}^*) + (\vec{M} - \vec{M}^*)^{\mathrm{T}}\mathbf{R}(\vec{M} - \vec{M}^*)]dt\,, \tag{21}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are quadratic positive-definite matrices of proper dimensions. Using the standard procedure we can find the gain matrix $\mathbf{K}$ (see for complete treatment of the LQR problem see [24]). The control action then takes the following form:

$$\vec{M} = -\mathbf{K}(\vec{s} - \vec{s}^*) + \vec{M}^* . \tag{22}$$

where $\vec{M}^* = \vec{M}^*(\vec{q}^*, \dot{\vec{q}}^*, \ddot{\vec{q}}^*)$ is the compensating control action that depends on the desired position of the system. It can be found by substituting desired values for generalized coordinates and their first two derivatives into (7), and then solving for $\vec{M}^*$:

$$\vec{M}^* = \mathbf{H}^{-1}[\mathbf{\Lambda}(\vec{q}^*)\ddot{\vec{q}}^* + \vec{C}(\vec{q}^*, \dot{\vec{q}}^*) + \vec{G}(\vec{q}^*) + \vec{\Phi}(\dot{\vec{q}}^*)] , \tag{23}$$

We cab observe that substituting (22) into (21) gives us another form of the cost function J:

$$J = \int (\vec{s} - \vec{s}^*)^{\mathrm{T}} (\mathbf{Q} + \mathbf{K}^{\mathrm{T}}\mathbf{R}\mathbf{K})(\vec{s} - \vec{s}^*) dt , \tag{24}$$

The equation (22) describes the work of the ILQR module on the diagram on the fig. 3.

5 Numerical simulation results

Let us first consider the case when the zero moment point P lies within the support polygon at the beginning of the motion. In that case only normal ZMP trajectory generation and ILQR controller are working during the motion. In the fig. 4 the time functions of actual and desired position of P, as well as horizontal displacement of the center of mass, are shown.

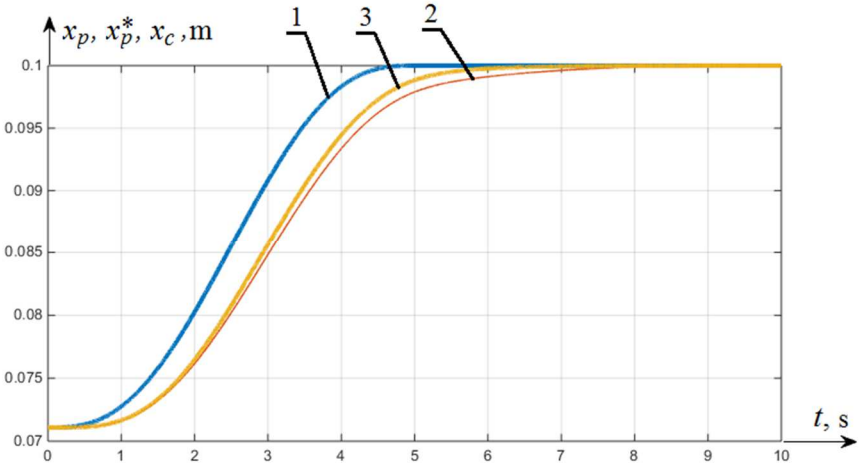

Fig. 4 Graphs of desired and actual position of point P for the case when verticalization; 1 – desired position of point P, 2 – actual position of point P, 3 – horizontal displacement of the center of mass

We should note that the actual position of the point P is different from its desired position because of control error and. The graphs in fig. 4 show that both desired and actual position of point P lies within interval $[0 \quad 0.2]$, which represents robot's support polygon in this planar case. As we can see the center of mass and the point P are moving in the same manner and their graphs are relatively similar. It suggests this case it would be possible to control position of the center of mass instead of point P (as it was done in [25, 26]).

To review the influence of the verticalization time on the performance of the controller let us consider the case when the sit-to-stand motion takes 2 seconds. In the fig. 5 the same graphs are shown for this case.

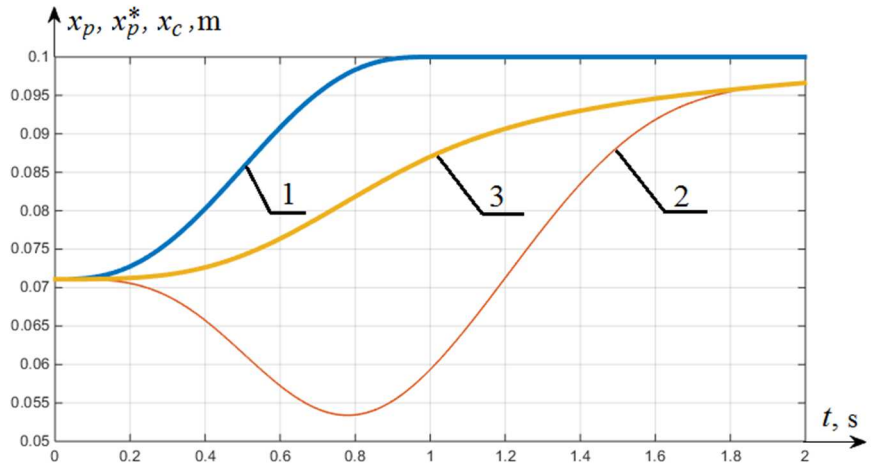

Fig. 5 Graphs of desired and actual position of point P for the case when verticalization; 1 – desired position of point P, 2 – actual position of point P, 3 – horizontal displacement of the center of mass

Graphs in the fig. 5 show that the center of mass and the point P are now moving in different way; the center of mass still monotonically moves towards the center of the support polygon, but the point P moves away from it for the first 0.8 seconds (40% of the duration of the motion). The fact that changing duration of the sit-to-stand process has this influence on the performance of the control system is explained by the fact that faster raise requires bigger accelerations of the joints, leading to bigger inertial forces.

We can note that the center of mass moves towards the center of the support polygon without oscillations. The fact that the generated desired value for the center of mass position has no oscillations allows the control system to generate desired time dependencies for generalized coordinates demonstrate relatively slow rate of change, which makes the feedback control easier.

6 Comparison with natural human motion

In this section we compare the motion of the system that uses ZMP controller with the natural human motion during verticalization. For the experimental studies a sensor system "ExoMeasurer" build in Southwest State University was used. ExoMeasurer is an exoskeleton that can be fixed to a human with straps and belts. The links have variable lengths, which allows to use the sensor system on individuals of different height. The system includes rotation sensors, accelerometers and pressure sensors, which allows to calculate the orientation and motion parameters of the exoskeleton, determine whether or not the feet have are in contact with the ground.

The questions of determining the positions of the centers of mass of different parts of human body are discussed in medical literature, here [27] was used as a reference.

On the figure 6 the trajectory of the center of mass of a human during sit-to-stand motion is shown.

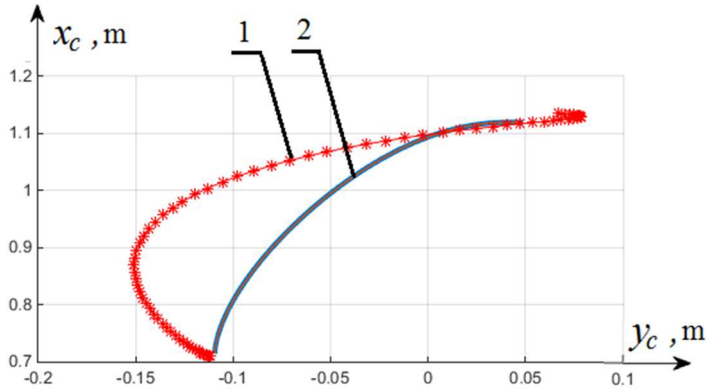

Fig. 6 Center of mass's trajectory; 1 – experimental results, 2 – motion of the exoskeleton obtained through simulation.

As we can see, the control system attempts to move the center of mass of the mechanism the same way as a human does during verticalization. The human's motion is characterized by wider motion of the center of mass, with visible oscillations. The center of mass trajectory produced by the ZMP based control system is more conservative and does not show oscillation.

7 Conclusion

In this paper a control system design for a lower limb exoskeleton was considered. To generate the trajectory of the center of mass of the exoskeleton ZMP methodology was used. Using the obtained desired trajectory for the center of mass the

trajectories for joint angles were produced using inverse kinematics. The feedback control was implemented with iterative linear quadratic regulator. Use of that regulator allowed to maintain small control error which made it possible to realize motion along the trajectory obtained from ZMP method. A drawback of the ILQR is the high computational intensity, which makes it necessary to use an onboard computer to do the computations.

In the paper the results of the numerical simulation of the exoskeleton motion are presented and analyzed. It is shown that the duration of the verticalization has significant influence of the behavior of the control system. Obtained results suggest that for a slower verticalization time it may not be necessary to control the position of the zero-moment point and that controlling horizontal displacement of the center of mass instead might be sufficient for maintaining vertical balance, because the two values behave in a similar manner. For faster sit-to-stand motion it is shown that the trajectory of the horizontal displacement of the center of mass significantly differs from the position of the zero moment point.

A comparison between natural human motion (for a human not wearing an exoskeleton) and the motion of a human in an exoskeleton is presented. It is shown that the trajectory of the center of mass of a human (without exoskeleton) during verticalization is similar to the same trajectory exoskeleton model that uses ZMP trajectory generation. At the same time natural human motion shows more oscillatory and less conservative behavior.

We should note that this study did not take into account inaccuracy of the sensors. This may limit the application of the obtained results for the cases when an exoskeleton is equipped with sensors that introduce significant disturbance in the feedback channel.

The future work includes studying the influence the update rate of ILQR has on the behavior of the control system and applying the proposed method to control a physical model of a lower limb exoskeleton.

Acknowledgment

Work is supported by RSF, Project № 14-39-00008

References

[1] Anam, Khairul, and Adel Ali Al-Jumaily. "Active exoskeleton control systems: State of the art." Procedia Engineering 41: 988-994 (2012)

[2] Contreras-Vidal, J.L. and Grossman, R.G. NeuroRex: A clinical neural interface roadmap for EEG-based brain machine interfaces to a lower body robotic exoskeleton. In Engineering in medicine and biology society (EMBC), 2013 35th annual international conference of the IEEE (pp. 1579-1582). IEEE (2013, July)

[3] Barbareschi, G., Richards, R., Thornton, M., Carlson, T. and Holloway, C. Statically vs dynamically balanced gait: Analysis of a robotic exoskeleton compared with a human. In Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE (pp. 6728-6731). IEEE (2015, August)

[4] Vukobratović, M. and Borovac, B. Zero-moment point—thirty five years of its life. International Journal of Humanoid Robotics, 1(01), pp.157-173 (2004)
[5] Kajita, S., Morisawa, M., Harada, K., Kaneko, K., Kanehiro, F., Fujiwara, K. and Hirukawa, H. Biped walking pattern generator allowing auxiliary zmp control. In Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on (pp. 2993-2999). IEEE (2006, October)
[6] Mitobe, K., Capi, G. and Nasu, Y. Control of walking robots based on manipulation of the zero moment point. Robotica, 18(06), pp.651-657 (2000)
[7] Choi, Y., You, B.J. and Oh, S.R. On the stability of indirect ZMP controller for biped robot systems. In Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on (Vol. 2, pp. 1966-1971). IEEE (2004, June)
[8] Low, K.H., Liu, X., Goh, C.H. and Yu, H. Locomotive control of a wearable lower exoskeleton for walking enhancement. Journal of Vibration and Control, 12(12), pp.1311-1336 (2006)
[9] G. Panovko, S. Savin, S. Jatsun, A. Yatsun Simulation of controlled motion of an exoskeleton in verticalization process Journal of Machinery Manufacture and Reliability, 2016 (in publishing).
[10] Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K. and Hirukawa, H. Biped walking pattern generation by using preview control of zero-moment point. In Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on (Vol. 2, pp. 1620-1626). IEEE (2003, September)
[11] Feng, S., Whitman, E., Xinjilefu, X. and Atkeson, C.G. Optimization-based Full Body Control for the DARPA Robotics Challenge. Journal of Field Robotics, 32(2), pp.293-312 (2015)
[12] Tsukahara, A., Hasegawa, Y., & Sankai, Y. Standing-up motion support for paraplegic patient with Robot Suit HAL. In Rehabilitation Robotics, 2009. ICORR 2009. IEEE International Conference on (pp. 211-217). IEEE (2009, June)
[13] Tsukahara, A., Kawanishi, R., Hasegawa, Y., & Sankai, Y. Sit-to-stand and stand-to-sit transfer support for complete paraplegic patients with robot suit HAL. Advanced robotics, 24(11), 1615-1638 (2010)
[14] Jun, H. G., Chang, Y. Y., Dan, B. J., Jo, B. R., Min, B. H., Yang, H., ... & Kim, J. Walking and sit-to-stand support system for elderly and disabled. In Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on (pp. 1-5). IEEE (2011, June)
[15] Taslim Reza, S. M., Ahmad, N., Choudhury, I. A., & Ghazilla, R. A. R. A Fuzzy Controller for Lower Limb Exoskeletons during Sit-to-Stand and Stand-to-Sit Movement Using Wearable Sensors. Sensors, 14(3), 4342-4363 (2014)
[16] Salah, O., Ramadan, A. A., Sessa, S., Ismail, A. A., Fujie, M., & Takanishi, A. Anfis-based sensor fusion system of sit-to-stand for elderly people assistive device protocols. International Journal of Automation and Computing,10(5), 405-413 (2013)
[17] Mughal, A. M., & Iqbal, K. 3D bipedal model for biomechanical sit-to-stand movement with coupled torque optimization and experimental analysis. In Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on (pp. 568-573). IEEE (2010, October)
[18] Jatsun S, Savin S, Yatsun A, Malchikov A. Study of Controlled Motion of Exoskeleton Moving from Sitting to Standing Position. In Advances in Robot Design and Intelligent Control 2016 (pp. 165-172). Springer International Publishing (2016)
[19] Jatsun S, Savin S, Yatsun A, Turlapov R. Adaptive control system for exoskeleton performing sit-to-stand motion. In Mechatronics and its Applications (ISMA), 10th International Symposium on 2015 Dec 8 (pp. 1-6). IEEE (2015)
[20] Jatsun S., Savin S., Yatsun A., Postolnyi A. Control system parameter optimization for lower limb exoskeleton with integrated elastic elements. Proceedings of the International Conference on CLAWAR 2016 (in publishing).
[21] Featherstone R. Rigid body dynamics algorithms. Springer; 10 (2014 November)
[22] Jatsun S. F., Vorochaeva L. Yu., Yatsun A. S., Savin S. I.. The modelling of the standing-up process of the anthropomorphic mechanism. Proceedings of the International Conference on CLAWAR. pp. 175-182 (2015).
[23] Li W, Todorov E. Iterative linear quadratic regulator design for nonlinear biological movement systems. InICINCO (1) 2004 Aug (pp. 222-229).
[24] Anderson, B.D. and Moore, J.B. Optimal control: linear quadratic methods. Courier Corporation (2007)
[25] Jatsun, S., Savin, S., Lushnikov, B. and Yatsun, A., 2016, January. Algorithm for motion control of an exoskeleton during verticalization. In ITM Web of Conferences (Vol. 6). EDP Sciences.

[26] Jatsun S., Savin S., Yatsun A. Parameter Optimization for Exoskeleton Control System Using Sobol Sequences Proceedings of 21st CISM-IFToMM Symposium on Robot Design 2016 (in publishing).
[27] Plagenhoef, S., Evans, F.G. and Abdelnour, T., 1983. Anatomical data for analyzing human motion. Research quarterly for exercise and sport, 54(2), pp.169-178.