EMITTER International Journal of Engineering Technology
Vol. 6, No. 1, June 2018
ISSN: 2443-1168

Walking Trajectory Optimization Algorithm For Robot Humanoid on Synthetic Grass

Dimas Pristovani R., Ardik Wijayanto, Ali Husein Alasiry, A. Subhan Khalilullah

Electronics Engineering Departement of Electronics Engineering Polytechnic Institute of Surabaya (EEPIS) and Multimedia Broadcasting Departement of Electronics Engineering Polytechnic Institute of Surabaya (EEPIS)
St. Raya ITS, Keputih, Sukolilo, Surabaya, 60111, Indonesia
E-mail: dimas@pasca.student.pens.ac.id, {ardik, ali, subhan}@pens.ac.id

Abstract

Synthetic grass surface is a new rule in international robot soccer competition (RoboCup). The main issue in the development of the RoboCup competition today is about how to make a humanoid robot walk above the field of synthetic grass. Because of that, the humanoid robot needs a system that can be implemented into the walking algorithm. This paper describes how to maintain the stability of humanoid robot called EROS by using walking trajectory algorithm without a control system. The establishment of the walking trajectory system is combined with a process of landing optimization using deceleration and heel-strikes gait optimization. This system has been implemented into a humanoid robot with 52 cm of height and walking on synthetic grass with different speeds. By adding optimization, the robot walks more stable from 32% to 80% of stability. In the next research, the control system will be added to improve the stability.

Keywords: humanoid robot, inverse kinematic, landing deceleration, heel-strikes gait, RoboCup, optimization algorithm.

1. INTRODUCTION

The kind of robots that resemble human body is humanoid robot. The humanoid robot has been made with a lot of purpose such as playing soccer. This kind of humanoid robot has competition called RoboCup [1]. On this Robocup competition, the robot must be able to adapt the human soccer player behavior. The main rules of the game are ball and goal have a white color, and the field has synthetic grass with green color [1]. The main issue in the development of the RoboCup competition today is about how to make this robot walk in stable condition (not fall) above the field of synthetic grass and how to identify the differences between ball, goal, and line of field. This paper will only discuss the problem

* Corresponding Author. Tel: +62-82337484588
E-mail: dimaspens@gmail.com

caused by synthetic grass to locomotion system of humanoid robot.

The humanoid robot in this discussion is called EROS. It has 20 Degrees of Freedom (DoF). With this 20 DoF, the Inverse Kinematic (IK) method is used to arrange the joint of legs and the other joint is arranged by using Forward Kinematic (FK) method [8]. The IK method is used to transform the Cartesian data space into joint data space, so the joint arrangement of legs would be easier. The input of IK method is obtained from walking trajectory system.

2. RELATED WORKS

Some researcher using trajectory system to define walking model of their humanoid robot. They use a polynomial equation or circle equation to generate the walking trajectory system [2]. The dynamics walking trajectory system has been proposed in [2,3,10,11], so the robot can adjust the walking trajectory system based on base surface (floor), external disturbance, and internal disturbance. The passive walking trajectory can also be used as an alternative to generate the walking trajectory system more dynamics [4]. The passive walking trajectory can reduce the energy consumption during walking locomotion. But, by using this model, the walking velocity is harder to improve than by using another model. In the other way to maintain the walking stability, researcher using control system. In the previous research, the trajectory joint was used to establish the walking trajectory in humanoid robot [5]. The method which used to arrange the acceleration and deceleration walking locomotion of EROS has been developed in previous research [6].

3. ORIGINALITY

As mentioned in the introduction, this paper only discus about the problem caused by synthetic grass filed without a control system. In previous research of EROS, the walking trajectory system is used in the flat field (green carpet). But in this new competition, the field was changed with a synthetic grass field. Because of it, when EROS is using previous walking trajectory system, the EROS always falling (the foot always stuck inside the synthetic grass surface). The idea to solve this problem is make the EROS robot walk like human. By using EROS system, the previous walking trajectory system is added with landing optimization with deceleration process [7] and heel-strikes gait optimization process during the process of Single Support Phase (SSP) to Double Support Phase (DSP). So, the EROS robot will walk look like more human.

The contribution of this paper is generating a new walking trajectory system that approaching human behavior. So, it will more adaptive with the synthetic grass field. Furthermore, this system has been applied into the EROS humanoid robot.

4. SYSTEM DESIGN

The system design of EROS humanoid robot is splited into 2 main section which is design of EROS humanoid robot including hardware construction,

mechanical construction, and kinematics design, and walking system including walking trajectory system, heel strikes gait optimization, landing optimization with deceleration, and comparation of implementation.

4.1 Design of EROS Humanoid Robot

This section will describe about design of EROS humanoid robot such as hardware construction, mechanical construction, and design kinematics.

A. Hardware Construction

In the Figure 1, there are 2 controller system called sub-controller and main-controller. The sub-controller serves as a motion controller, collect and processing sensor data, and supply management in the EROS humanoid robot. In the main-controller serves as processing the vision from camera sensor and make the artificial intelligent. The main-controller also controlling the sub-controller by using serial communication.

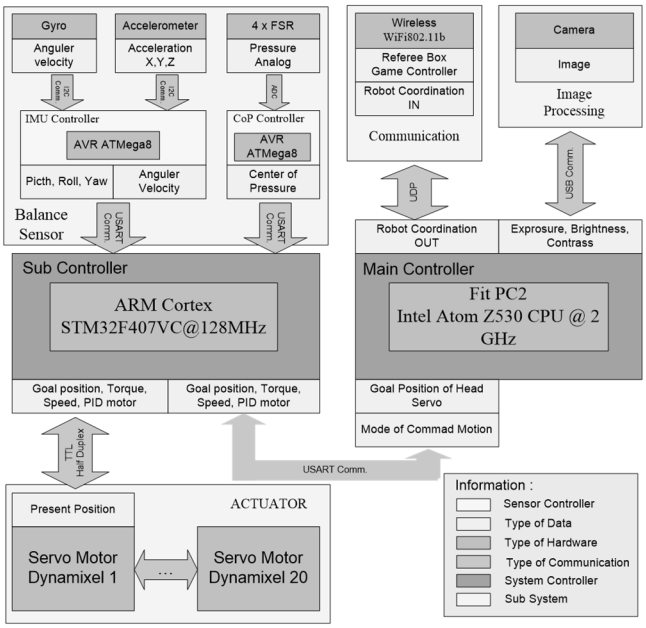

Figure 1. Hardware construction of EROS humanoid robot

B. Mechanical Construction

The mechanical construction of EROS humanoid robot can be seen in Figure 2. The mechanical structure is made of aluminum with high precision. EROS humanoid robot has 20 DoF and 52 cm of height [9]. The mechanical structure was made by the existing rule of RoboCup competition [1]. The distribution of degree of freedom is 12 DoF in the legs, 6 DoF in the arms, 2 in the head. The configuration of DoF can be seen in Table 1.

Table 1. Channel clasification in mobile communication

Joint	Degree of Freedom (DoF)
Head	2
Right Arm	3
Left Arm	3
Right Leg	6
Left Leg	6
Total	20

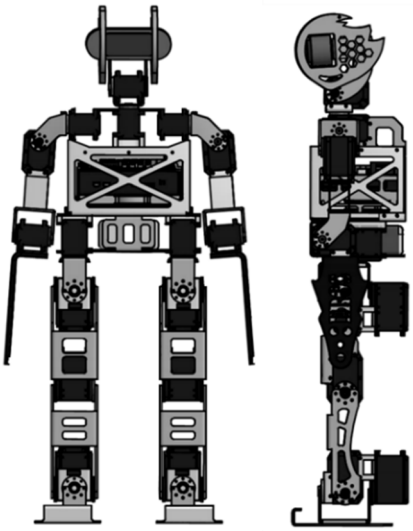

Figure 2. Mechanical design of EROS humanoid robot

B. Kinematics Design

With 20 DoF in the EROS humanoid robot, then the kinematic analysis is divided into several parts: head, arm, and legs. The head and arm parts are analyzed by using FK method. The leg is analyzed by using IK method.

B.1 Head Kinematic

In the head kinematic, the FK method is used to analyze the head mechanism. The design kinematic rotational joint of head mechanism can be seen in Figure 3 and the configuration of each joint can be seen in Table 2.

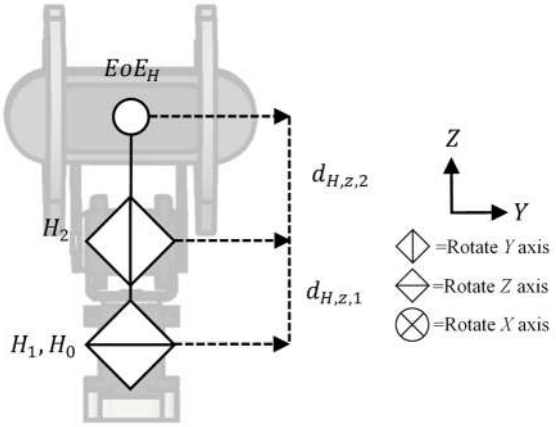

Figure 3. Mechanical design of EROS humanoid robot

Table 2. Configuration of each joint in the head

Joint	Rotational	Translation		
		x	***y***	***z***
H_1	$R_z(\theta_{H,1})$	0	0	$d_{H,z,1}$
H_2	$R_y(\theta_{H,2})$	0	0	$d_{H,z,2}$

By using homogenous transformation, the transformation of each joint is shown in equation below.

$$ {}^{0}_{2}T_H = {}^{0}_{1}T_H . {}^{1}_{2}T_H \tag{1} $$

The total transformation of head mechanism is shown in equation below.

$$ {}^{0}_{2}T_H = \begin{bmatrix} \cos\theta_{H,1}\cos\theta_{H,2} & -\sin\theta_{H,1} & \cos\theta_{H,1}\sin\theta_{H,2} & d_{H,z,2}\cos\theta_{H,1}\sin\theta_{H,2} \\ \cos\theta_{H,2}\sin\theta_{H,1} & \cos\theta_{H,1} & \sin\theta_{H,1}\sin\theta_{H,2} & d_{H,z,2}\sin\theta_{H,2}\sin\theta_{H,2} \\ -\sin\theta_{H,2} & 0 & \cos\theta_{H,2} & d_{H,z,1}+d_{H,z,2}\cos\theta_{H,2} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2} $$

From equation (2), the position vector of End of Effector Head (EoE_H) is calculated by using equation below.

$$ \begin{pmatrix} P_{H,x} \\ P_{H,y} \\ P_{H,z} \end{pmatrix} = \begin{bmatrix} d_{H,z,2}\cos\theta_{H,1}\sin\theta_{H,2} \\ d_{H,z,2}\sin\theta_{H,2}\sin\theta_{H,2} \\ d_{H,z,1}+d_{H,z,2}\cos\theta_{H,2} \end{bmatrix} \tag{3} $$

Where ${}^{0}_{1}T_H$ and ${}^{1}_{2}T_H$is homogenous transformation from rotational and translational joint $H_1$and H_2, $\theta_{H,1}$ and $\theta_{H,2}$ is rotational angle jointH_1andH_2,

$d_{H,z,1}$ and $d_{H,z,2}$ is translational vector from present joint to next joint. ${}^{0}_{2}T_{H}$ is homogenous transformation from EoE_H, $P_{H,(x,y,z)}$ is position vector of EoE_H.

B.2 Left and Right Arms Kinematic

In the left and right arms kinematic, the FK method is used to analyze the left and right arms mechanism. The left and right arms mechanism has the same structure. Design kinematic rotational joint of right arm can be seen in Figure 4 and the configuration of each joint can be seen in Table 3. Design kinematic rotational joint of left arm can be seen in Figure 5 and the configuration of each joint can be seen in Table 4.

Table 3. Configuration of each joint in the right arm

Joint	Rotational	Translation		
		x	***y***	***z***
RA_1	$R_y(\theta_{RA,1})$	0	$d_{RA,y,1}$	$-d_{RA,z,1}$
RA_2	$R_x(\theta_{RA,2})$	0	0	$-d_{RA,z,2}$
RA_3	$R_y(\theta_{RA,3})$	0	0	$-d_{RA,z,3}$

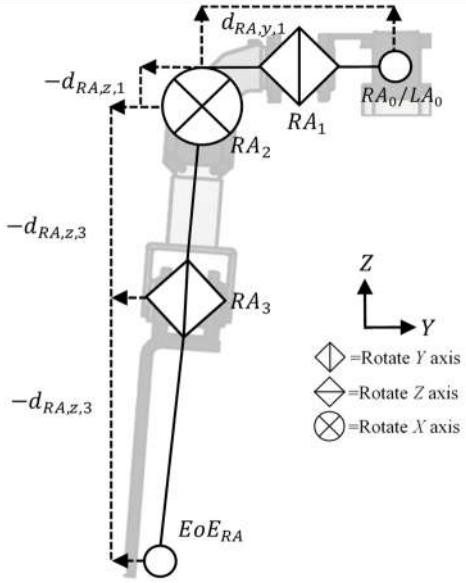

Figure 4. Kinematic rotational joint of right arm mechanism

By using homogenous transformation, the transformation of each joint is shown in equation below.

$$ {}^{0}_{3}T_{RA} = {}^{0}_{1}T_{RA}.\,{}^{1}_{2}T_{RA}.\,{}^{2}_{3}T_{RA} \quad (4) $$

The total transformation of right arm mechanism is shown in equation below.

$$ {}^{0}_{3}T_{RA} = \begin{bmatrix} R_{RA,11} & R_{RA,12} & R_{RA,13} & R_{RA,14} \\ R_{RA,21} & R_{RA,22} & R_{RA,23} & R_{RA,24} \\ R_{RA,31} & R_{RA,32} & R_{RA,33} & R_{RA,34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5) $$

Where:

$R_{RA,11} = \cos\theta_{RA,1}\cos\theta_{RA,3} - \cos\theta_{RA,2}\sin\theta_{RA,1}$

$R_{RA,21} = \sin\theta_{RA,2}\sin\theta_{RA,3}$

$R_{RA,31} = -\cos\theta_{RA,3}\sin\theta_{RA,1} - \cos\theta_{RA,1}\cos\theta_{RA,2}\sin\theta_{RA,3}$

$R_{RA,12} = \sin\theta_{RA,1}\sin\theta_{RA,2}$

$R_{RA,22} = \cos\theta_{RA,2}$

$R_{RA,32} = \cos\theta_{RA,1}\sin\theta_{RA,2}$

$R_{RA,13} = \cos\theta_{RA,1}\sin\theta_{RA,3} + \cos\theta_{RA,3}\cos\theta_{RA,2}\sin\theta_{RA,1}$

$R_{RA,23} = -\cos\theta_{RA,3}\sin\theta_{RA,2}$

$R_{RA,33} = \cos\theta_{RA,1}\cos\theta_{RA,2}\cos\theta_{RA,3} - \sin\theta_{RA,1}\sin\theta_{RA,3}$

$R_{RA,14} = -d_{RA,z,1}\sin\theta_{RA,1} - \sin\theta_{RA,1}\emptyset_{RA,1}$

$-d_{RA,z,3}\cos\theta_{RA,1}\sin\theta_{RA,3}$

$R_{RA,24} = d_{RA,x,1} + d_{RA,z,2}\sin\theta_{RA,2} + d_{RA,z,3}\cos\theta_{RA,3}\sin\theta_{RA,2}$

$R_{RA,34} = d_{RA,z,3}\sin\theta_{RA,1}\sin\theta_{RA,3}$

$-\cos\theta_{RA,1}\emptyset_{RA,1} - d_{RA,z,1}\cos\theta_{RA,1}$

$\emptyset_{RA,1} = d_{RA,z,2}\cos\theta_{RA,2} + d_{RA,z,3}\cos\theta_{RA,2}\cos\theta_{RA,3}$

From equation (5), the position vector of End of Effector Right Arm (EoE_{RA}) is calculated by using equation below.

$$ \begin{pmatrix} P_{RA,x} \\ P_{RA,y} \\ P_{RA,-z} \end{pmatrix} = \begin{bmatrix} R_{RA,14} \\ R_{RA,24} \\ R_{RA,34} \end{bmatrix} \quad (6) $$

Where ${}^{0}_{1}T_{RA}$, ${}^{1}_{2}T_{RA}$ and ${}^{2}_{3}T_{RA}$ is homogenous transformation from rotational and translational joint RA_1, RA_2, and RA_3, $\theta_{RA,1}$, $\theta_{RA,2}$ and $\theta_{RA,3}$ is rotational angle joint RA_1, RA_2, and RA_3. $d_{RA,x,1}$, $d_{RA,z,1}$, $d_{RA,z,2}$, and $d_{RA,z,3}$ is translational vector from present joint to next joint. ${}^{0}_{3}T_{RA}$ is homogenous transformation from EoE_{RA}, $P_{RA,(x,y,z)}$ is position vector of EoE_{RA}. Important to remember that position vector in Z axis is always negative when the right arm in normal position ($\theta_{RA,1,2,3} = 0$), because the translation of joint in Z axis is always negative, so the $P_{RA,(-z)}$ in normal position is always has negative value.

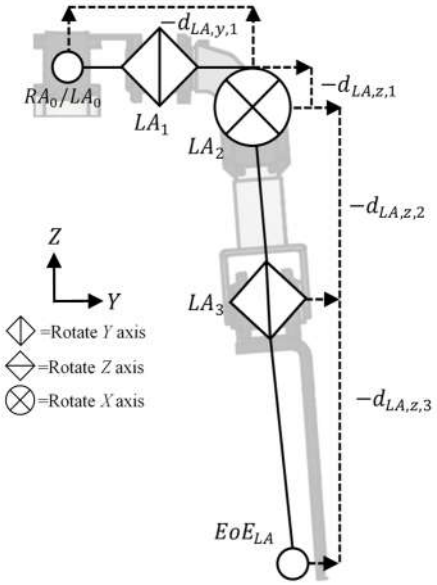

Figure 5. Kinematic rotational joint of left arm mechanism

Table 4. Configuration of each joint in the left arm

Joint	Rotational	Translation		
		x	***y***	***z***
LA_1	$R_y(\theta_{LA,1})$	0	$-d_{LA,y,1}$	$-d_{LA,z,1}$
LA_2	$R_x(\theta_{LA,2})$	0	0	$-d_{LA,z,2}$
LA_3	$R_y(\theta_{LA,3})$	0	0	$-d_{LA,z,3}$

The differences of left and right arms are the translational vector of joint 1 (LA_1 and RA_1) in *Y* axis. So, in the left arm, the transformation of each joint is like transformation of each joint in right arm, just change the indexing name. By using right arm total transformation, the left arm transformation is shown in equation below.

$$ {}^{0}_{3}T_{LA} = \begin{bmatrix} R_{LA,11} & R_{LA,12} & R_{LA,13} & R_{LA,14} \\ R_{LA,21} & R_{LA,22} & R_{LA,23} & R_{LA,24} \\ R_{LA,31} & R_{LA,32} & R_{LA,33} & R_{LA,34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7) $$

Where:

$R_{LA,11}{\sim}R_{LA,31} = R_{RA,11}{\sim}R_{RA,31}$

$R_{LA,12}{\sim}R_{LA,32} = R_{RA,12}{\sim}R_{RA,32}$

$R_{LA,13}{\sim}R_{LA,33} = R_{RA,13}{\sim}R_{RA,33}$

$R_{LA,14}{\sim}R_{LA,34} = R_{RA,14}{\sim}R_{RA,34}$

$R_{LA,24} = -d_{LA,x,1} + d_{LA,z,2}\sin\theta_{LA,2} + d_{LA,z,3}\cos\theta_{LA,3}\sin\theta_{LA,2}$

$\emptyset_{LA,1} = \emptyset_{RA,1}$

From equation (7), the position vector of End of Effector Right Arm (EoE_{RA}) is calculated by using equation below.

$$\begin{pmatrix} P_{LA,x} \\ P_{LA,y} \\ P_{LA,-z} \end{pmatrix} = \begin{bmatrix} R_{LA,14} \\ R_{LA,24} \\ R_{LA,34} \end{bmatrix} \quad (8)$$

Where $\theta_{LA,1}$, $\theta_{LA,2}$ and $\theta_{LA,3}$ is rotational angle joint LA_1, LA_2, and LA_3, $d_{LA,x,1}$, $d_{LA,z,1}$, $d_{LA,z,2}$, and $d_{LA,z,3}$ is translational vector from present joint to next joint. ${}^{0}_{3}T_{LA}$ is homogenous transformation from EoE_{LA}, $P_{LA,(x,y,z)}$ is position vector of EoE_{LA}. Important to remember that position vector in Z axis is always negative when the left arm in normal position ($\theta_{LA,1,2,3} = 0$), because the translation of joint in Z axis is always negative, so the $P_{LA,(-z)}$ in normal position is always has negative value.

B.3 Left and Right Legs Kinematic

In the left and right legs kinematic, the IK method is used to analyze the left and right legs mechanism. The left and right legs mechanism has the same structure. If one of them has been analyzed, the analysis will represent the other. The design kinematic rotational joint of right leg mechanism can be seen in Figure 6 and the configuration of each joint can be seen in Table 5. The design kinematic rotational joint of left leg mechanism can be seen in Figure 10 and the configuration of each joint can be seen in Table 6.

The calculation of IK method is done by using complex trigonometry and reflection method. The IK analysis of legs is divided into 3 parts of analysis. The first part is analysis based on transverse plane (*X, Y*), the second part is analysis based on frontal plane (*Y, Z*), and the third part is analysis based on sagittal plane (*X, Z*).

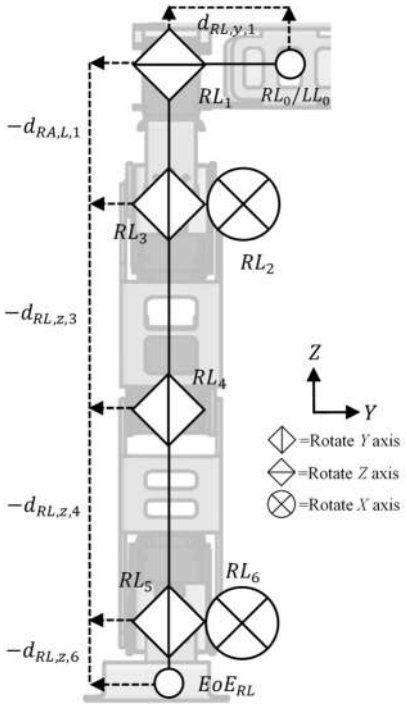

Figure 6. Kinematic rotational joint of right leg mechanism

Table 5. Configuration of each joint in the right leg

Joint	Rotational	Translation		
		x	***y***	***z***
RL_1	$R_z(\theta_{RL,1})$	0	$d_{RL,y,1}$	$-d_{RL,z,1}$
RL_2	$R_x(\theta_{RL,2})$	0	0	0
RL_3	$R_y(\theta_{RL,3})$	0	0	$-d_{RL,z,3}$
RL_4	$R_y(\theta_{RL,2})$	0	0	$-d_{RL,z,4}$
RL_5	$R_y(\theta_{RL,3})$	0	0	0
RL_6	$R_x(\theta_{RL,3})$	0	0	$-d_{RL,z,6}$

The IK is using to transform the cartesian space into joint space. The cartesian space data of the right leg is position vector of EoE_{RL} ($P_{RL,(x,y,-z)}$) and the orientation angel ($\theta_{RL,Heading}$). Important to remember that position vector in Z axis is always negative when the right leg in normal position ($\theta_{RL,1,2,3,4,5,6} = 0$) because the translation of joint in Z axis is always negative, so the $P_{RL,(-z)}$ in normal position is always has negative value. The beginning to analysis the IK is find the relation between x and y position (transverse plane), it can be seen in Figure 7 and the calculation is shown in equation below.

$$\theta_{RL,1} = \theta_{RL,Heading} \tag{9}$$

$$R0_{RL} = \sqrt{P_{RL,(x)}{}^2 + P_{RL,(y)}{}^2} \tag{10}$$

$$\theta_{RL,\beta} = \tan^{-1}\left({}^{P_{RL,(y)}}\!/_{P_{RL,(x)}}\right) - \theta_{RL,1} \quad (11)$$

$$\Delta P_{RL,(y)} = P_{RL,(y)} + \left(R0_{RL} \times \sin\theta_{RL,\beta}\right) \quad (12)$$

$$\Delta P_{RL,(x)} = P_{RL,(x)} - \left(R0_{RL} \times \cos\theta_{RL,\beta}\right) \quad (13)$$

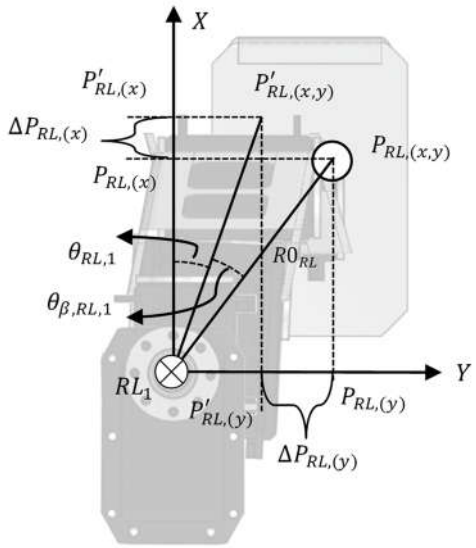

Figure 7. Right leg kinematic model from transverse plane view (top view)

Where $\theta_{RL,1}$ is joint angle in the RL_1 joint. The $\theta_{RL,1}$ is special joint because the input is directly from joint space and become a reference of the leg orientation. $R0_{RL}$ is resultant that occurs because existence of $P_{RL,(x)}$ and $P_{RL,(y)}$. $\theta_{RL,\beta}$is angle deviation between present position vector of EoE_{RL} ($P_{RL,(x,y)}$) with the next position vector of EoE_{RL} ($P'_{RL,(x,y)}$). $\Delta P_{RL,(x)}$ and $\Delta P_{RL,(y)}$ is distance deviation between present position vector of EoE_{RL} ($P_{RL,(x,y)}$) with the next position vector of EoE_{RL} ($P'_{RL,(x,y)}$). $\Delta P_{RL,(x)}$ and $\Delta P_{RL,(y)}$ is used for IK calculation in the frontal plane of IK analysis, $P_{RL,(x)}$ is replaced by $\Delta P_{RL,(x)}$ and $P_{RL,(y)}$ is replaced by $\Delta P_{RL,(y)}$. Furthermore, the analysis of IK is finding the relation between *Y* and *Z* position (frontal plane). It can be seen in Figure 8 and the calculation is shown in equation below.

$$P_{RL_6,(-z)} = P_{RL,(-z)} - \left(-d_{RL,z,6}\right) \quad (14)$$

$$\theta_{RL,2} = \tan^{-1}\left({}^{\Delta P_{RL,(y)}}\!/_{P_{RL_6,(-z)}}\right) \quad (15)$$

$$R1_{RL} = \sqrt{\Delta P_{RL,(y)}{}^2 + P_{RL_6,(-z)}{}^2} \quad (16)$$

Where $P_{RL_6,(-z)}$ is new height for IK calculation, because the joint RL_6 has a translation of $-d_{RL,z,6}$ from EoE_{RL}. $\theta_{RL,2}$ and $\theta_{RL,6}$ is joint angle in the joint RL_2 (Hip roll angle) and RL_6 (Ankle roll angle). $R1_{RL}$is resultant that occurs because existence of $P_{RL_6,(-z)}$and $\Delta P_{RL,(y)}$.

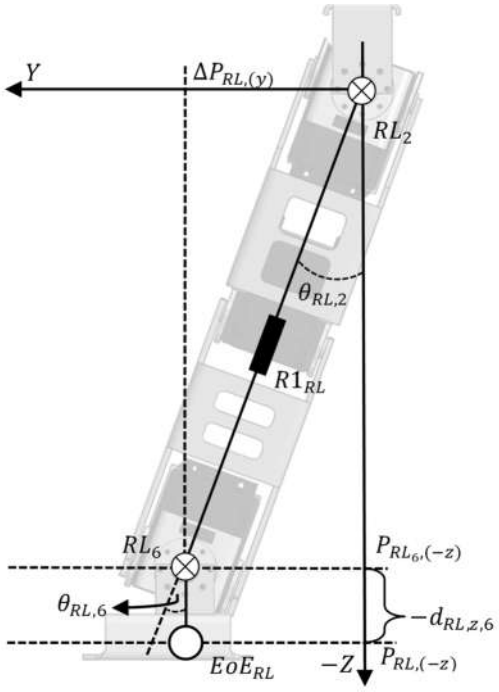

Figure 8. Right leg kinematic model from frontal plane view (front view)

$R1_{RL}$ is used to arrange new height of the leg when analyzing IK in the sagittal plane. Moreover, the analysis of IK is finding the relation between X and Z position (sagittal plane), it can be seen in Figure 9 and the calculation is shown in equation below.

$$R2_{RL} = \sqrt{\Delta P_{RL,(x)}{}^2 + P_{RL_6,(-z)}{}^2} \quad (17)$$

$$S_{RL,\gamma c,a} = \left(\left(-d_{RL,z,3}\right)^2 + \left(-d_{RL,z,4}\right)^2 - R2_{RL}{}^2\right) \quad (18)$$

$$S_{RL,\gamma c,b} = \left(2 \times \left(-d_{RL,z,3}\right) \times \left(-d_{RL,z,4}\right)\right) \quad (19)$$

$$\theta_{RL,\gamma c} = \cos^{-1}\left(S_{RL,\gamma c,a} \Big/ S_{RL,\gamma c,b}\right) \quad (20)$$

$$\theta_{RL,\gamma a} = \tan^{-1}\left(\Delta P_{RL,(x)} \Big/ R1_{RL}\right) \quad (21)$$

$$S_{RL,\gamma b,a} = \left(\left(-d_{RL,z,3}\right) \times \sin\theta_{RL,\gamma c}\right) \quad (22)$$

$$S_{RL,\gamma b,b} = \left(\left(-d_{RL,z,3}\right) + \left(\left(-d_{RL,z,4}\right) \times \cos\theta_{RL,\gamma c}\right)\right) \quad (23)$$

$$\theta_{RL,\gamma b} = \tan^{-1}\left(S_{RL,\gamma b,a} \Big/ S_{RL,\gamma b,b}\right) \quad (24)$$

$$\theta_{RL,3} = \theta_{RL,\gamma a} + \theta_{RL,\gamma b} \quad (25)$$

$$\theta_{RL,4} = 180 - \theta_{RL,\gamma c} \quad (26)$$

$$\theta_{RL,5} = \theta_{RL,3} + \theta_{RL,4} \quad (27)$$

$$\theta_{RL,6} = -\theta_{RL,2} \quad (28)$$

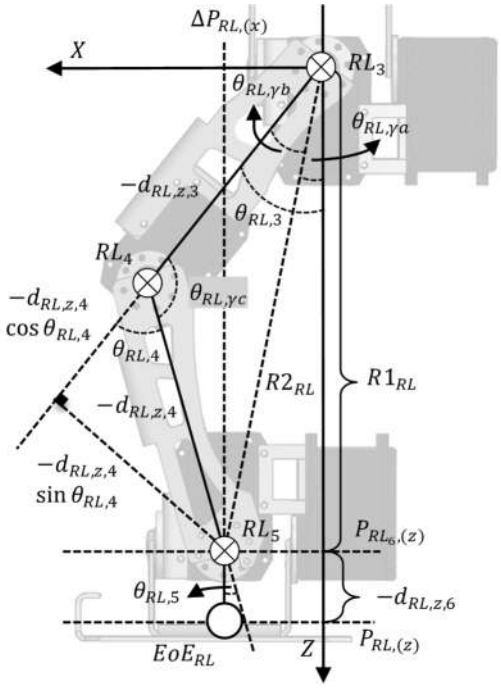

Figure 9. Right leg kinematic model from sagittal plane (side view)

Where $R2_{RL}$ is resultant from $\Delta P_{RL,(x)}$ and $P_{RL_6,(-z)}$, this resultant is used to calculate $\theta_{RL,\gamma c}$. $\theta_{RL,\gamma c}$ is inside knee angle, this angle is calculated by using $P_{RL_6,(-z)}$, $-d_{RL,z,4}$, and $-d_{RL,z,3}$. $\theta_{RL,3}$ is RL_3 joint angle (Hip pitch angle). $\theta_{RL,3}$ divided into two parts called $\theta_{RL,\gamma a}$ and $\theta_{RL,\gamma b}$. The $\theta_{RL,\gamma a}$ is calculated by using trigonometry with $\Delta P_{RL,(x)}$ and $R1_{RL}$. The $\theta_{RL,\gamma b}$ is calculated by using trigonometry with $\left(-d_{RL,z,3}\right)\sin\theta_{RL,\gamma c}$ and $\left(-d_{RL,z,3}\right)+\left(-d_{RL,z,4}\right)\cos\theta_{RL,\gamma c}$. $\theta_{RL,4}$ is RL_4 joint angle (Knee pitch angle). $\theta_{RL,4}$ is calculated from $180-\theta_{RL,\gamma c}$. $\theta_{RL,5}$ is RL_5 joint angle (Ankle pitch angle). $\theta_{RL,5}$ is calculated from sum of $\theta_{RL,3}$ and $\theta_{RL,4}$. $\theta_{RL,6}$ is RL_6 joint angle (Ankle roll angle). $\theta_{RL,6}$ is calculated from reflection angle of Hip roll angle ($-\theta_{RL,2}$). The origin joint (LL_0 and RL_0) has similar position vector. The differences of left and right legs are the translational vector of joint 1 (LL_1 and RL_1) in Y axis from origin position vector (LL_0 and RL_0). It is like differences between left and right arm.

Table 6. Configuration of each joint in the right leg

Joint	Rotational	Translation		
		x	y	z
LL_1	$R_z(\theta_{LL,1})$	0	$-d_{LL,y,1}$	$-d_{LL,z,1}$
LL_2	$R_x(\theta_{LL,2})$	0	0	0
LL_3	$R_y(\theta_{LL,3})$	0	0	$-d_{LL,z,3}$
LL_4	$R_y(\theta_{LL,2})$	0	0	$-d_{LL,z,4}$
LL_5	$R_y(\theta_{LL,3})$	0	0	0
LL_6	$R_x(\theta_{LL,3})$	0	0	$-d_{LL,z,6}$

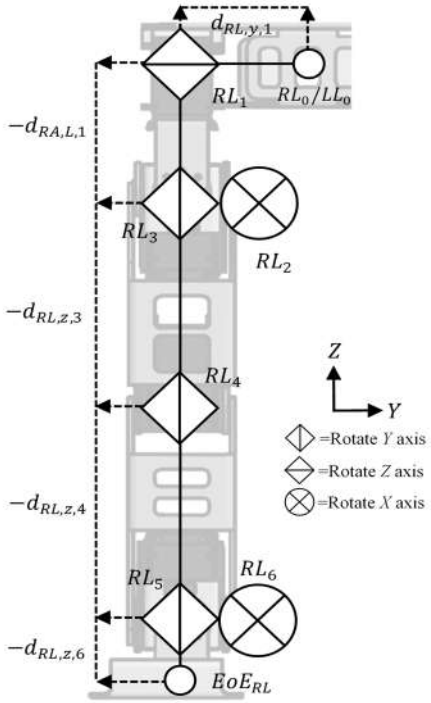

Figure 10. Kinematic rotational joint of left leg mechanism

So, in the left leg, the IK equation model of each joint is similar to IK equation model of each joint in the right leg, just change the indexing name. By using right leg total IK equation model, the left leg IK equation is shown in the equation below.

$$\theta_{LL,1} = \theta_{LL,Heading} \tag{29}$$

$$\theta_{LL,2} = \tan^{-1}\left(\Delta P_{LL,(y)} \Big/ P_{LL_6,(-z)}\right) \tag{30}$$

$$\theta_{LL,\gamma(a,b,c)} = \theta_{RL,\gamma(a,b,c)} \tag{31}$$

$$\theta_{LL,3} = \theta_{LL,\gamma a} + \theta_{LL,\gamma b} \tag{32}$$

$$\theta_{LL,4} = 180 - \theta_{LL,\gamma c} \tag{33}$$

$$\theta_{LL,5} = \theta_{LL,3} + \theta_{LL,4} \tag{34}$$

$$\theta_{LL,6} = -\theta_{LL,2} \tag{35}$$

Where $\theta_{RL,1}$ is joint angle in the RL_1 joint. $\theta_{RL,2}$ is joint angle in the joint RL_2 (Hip roll angle). $\theta_{LL,\gamma(a,b,c)}$ is angle support to calculate $\theta_{RL,3}$ and $\theta_{RL,4}$. $\theta_{RL,3}$ is RL_3 joint angle (Hip pitch angle). $\theta_{RL,4}$ is RL_4 joint angle (Knee pitch angle). $\theta_{RL,5}$ is RL_5 joint angle (Ankle pitch angle). $\theta_{RL,6}$ is joint angle in the joint RL_6 (Ankle roll angle).

4.2 Walking System

This section will describe about walking system of EROS humanoid robot such as walking trajectory system, landing optimization with deceleration,

heel-strikes gait optimization, and comparison of implementation (before and after optimization).

A. Walking Trajectory System

Walking trajectory system is discussed into two parts, the walking trajectory sagittal plane view (X and Z axis) in Figure 11 and the walking trajectory frontal plane view (Y and Z axis) in Figure 12. Walking trajectory system has 4 steps trajectory for 1 cycle of walk.

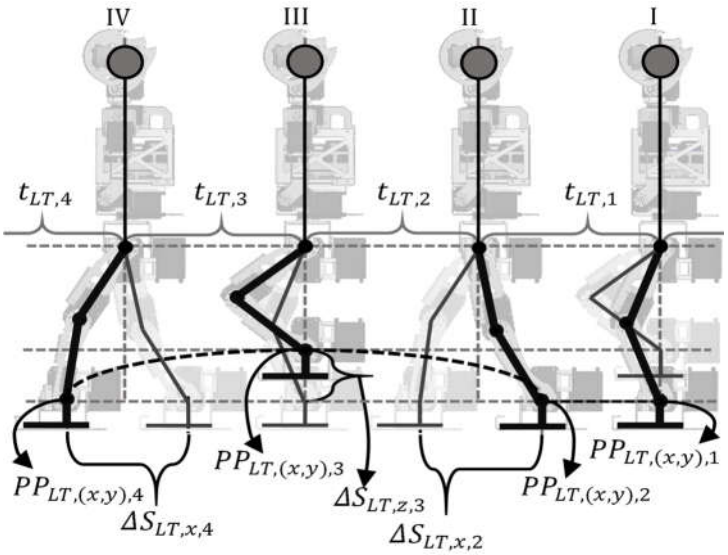

Figure 11. Walking trajectory sagittal plane view

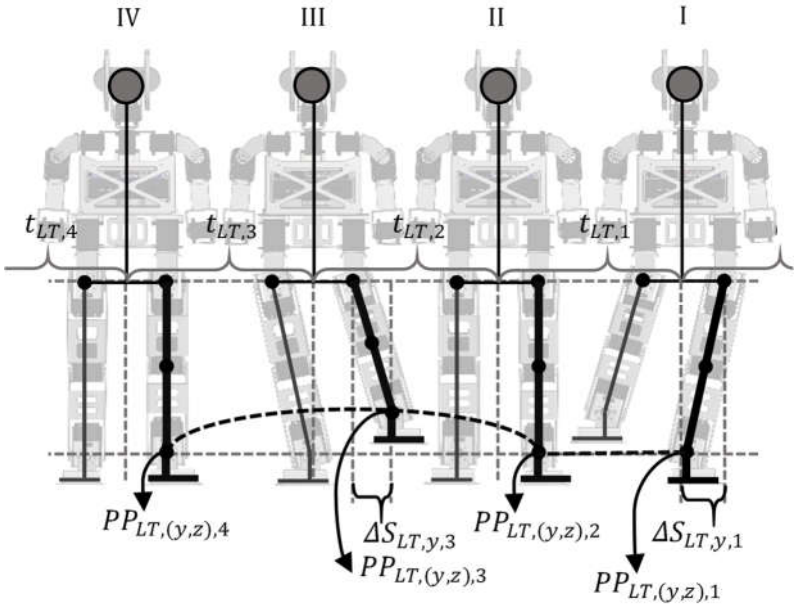

Figure 12. Walking trajectory frontal plane view

The first step is Single Support Phase (SSP) to Double Support Phase (DSP) for right leg, the second step is DSP to SSP for left leg, the third step is SSP to DSP for left leg, and the fourth is DSP to SSP for right leg.

In Figure 11 and Figure 12 explain the left leg step in one cycle of walk. The left and right leg is using similar system. By using circle equation, the input of equation will not in the time domain, but always in angle domain. In this discussion, walking trajectory system is generated by using quarter circle equation in each step. The input is on 0 ~ 90 degree in each transition step. The result from calculation of quarter circle equation is points of trajectory. These points is used and combined to generate the trajectory system. It is shown in Figure 13 and Figure 14. This figure is explaining left leg walking trajectory system (the index is *LT*).

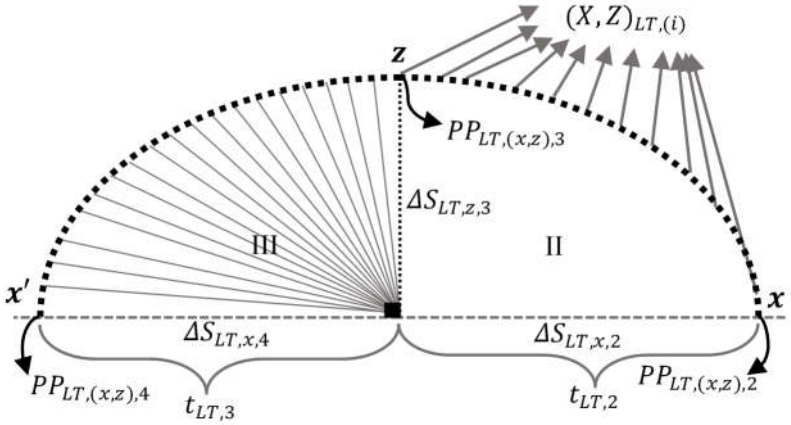

Figure 13. Walking trajectory generator result sagital plane view

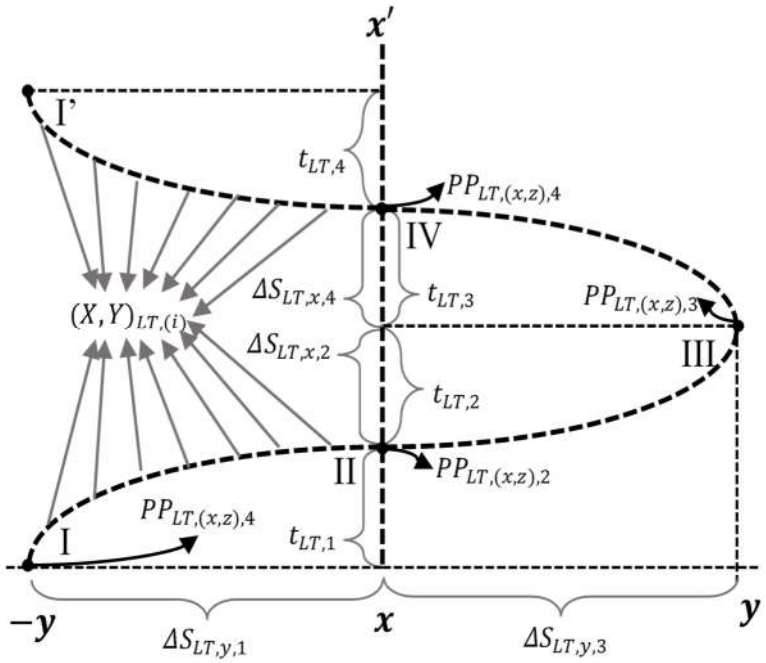

Figure 14. Walking trajectory generator result transversal plane view

The transition of each step will spend time about $t_{LT,(i)}$. This time depend on transition of each point in trajectory system that is used. Because of that, the transition arrangement between each point will influence to the speed of walk. The left (LT) and right (RT) legs equation of quarter circle in Z axis, X axis, and Y axis is shown in the equation bellow.

$$\Delta S_{LT,(x,y,z),(i)} = PP_{LT,(x,y,z),(i+1)} - PP_{LT,(x,y,z),(i)} \quad (36)$$

$$X_{LT,(T_T+T_{mn})} = \left((-\cos(T_T + T_{mn}) + 1) \times \Delta S_{LT,x,(i)}\right) + PP_{LT,x,(i)} \quad (37)$$

$$(Y,Z)_{LT,(2,4),(T_T+T_{mn})} = \left(\sin(T_T + T_{mn}) \times \Delta S_{LT,(y,z),(i)}\right) + PP_{LT,(y,z),(i)} \quad (38)$$

$$(Y,Z)_{LT,(1,3),(T_T+T_{mn})} = \left((1 - \sin(T_T + T_{mn}) + 90) \times \Delta S_{LT,(y,z),(i)}\right) + PP_{LT,(y,z),(i)} \quad (39)$$

$$\Delta S_{RT,(x,y,z),(i)} = PP_{RT,(x,y,z),(i+1)} - PP_{RT,(x,y,z),(i)} \quad (40)$$

$$X_{RT,(T_T+T_{mn})} = \left((-\cos(T_T + T_{mn}) + 1) \times \Delta S_{RT,x,(i)}\right) + PP_{RT,x,(i)} \quad (41)$$

$$(Y,Z)_{RT,(2,4),(T_T+T_{mn})} = \left(\sin(T_T + T_{mn}) \times \Delta S_{RT,(y,z),(i)}\right) + PP_{RT,(y,z),(i)} \quad (42)$$

$$(Y,Z)_{RT,(1,3),(T_T+T_{mn})} = \left((1 - \sin(T_T + T_{mn}) + 90) \times \Delta S_{RT,(y,z),(i)}\right) + PP_{RT,(y,z),(i)} \quad (43)$$

Where $PP_{LT,(x,y,z),(i)}$ and $PP_{LT,(x,y,z),(i+1)}$ is left leg present point and future point reference. $PP_{RT,(x,y,z),(i)}$ and $PP_{RT,(x,y,z),(i+1)}$ is right leg present point and future point reference $\Delta S_{(RT,LT),(x,y,z),(i)}$ is difference from present point reference $PP_{(RT,LT),(x,y,z),(i)}$ and future point reference $PP_{(RT,LT),(x,y,z),(i+1)}$ between steps. $(X,Y,Z)_{(RT,LT),(i),(T_T+T_{mn})}$ is points of trajectory. $(T_T + T_{mn})$ is input from degree space. The start value of this input is always zero (0) degree and the end value is always 90 degrees ($0 \leq T_T \leq 90$). This input will always have summed by T_{mn} ($T_{T(i)} = T_{T(i-1)} + T_{mn}$). So, each transition between present input and last input has a difference of T_{mn}.

B. Heel-Strikes Gait Optimization

Heel-strikes gait was adapted from human walking behavior to optimize the walking trajectory system. Heel-strikes gait is controlling ankle joint when beginning of transition process from DSP to SSP state (Push-Off) and SSP to DSP state (Collision).

$$\Delta S_{(RL,LL),H,z} = Z_{(RT,LT),(i)} - Z_{(LT,RT),(i)} \quad (44)$$

$$\Delta S_{(RL,LL),H,z}{}' = Z_{(RT,LT),(i)}{}' - Z_{(LT,RT),(i)}{}' \quad (45)$$

$$\theta_{(RL,LL),H,\alpha} = \left({\Delta S_{(RL,LL),H,z}}/{C_{H,\alpha\beta e}}\right) + \theta_{(RL,LL),4} \quad (46)$$

$$\theta_{(RL,LL),H,\beta} = \left({\Delta S_{(RL,LL),H,z}{}'}/{C_{H,\alpha\beta e}}\right) + \theta_{(RL,LL),4}{}' \quad (47)$$

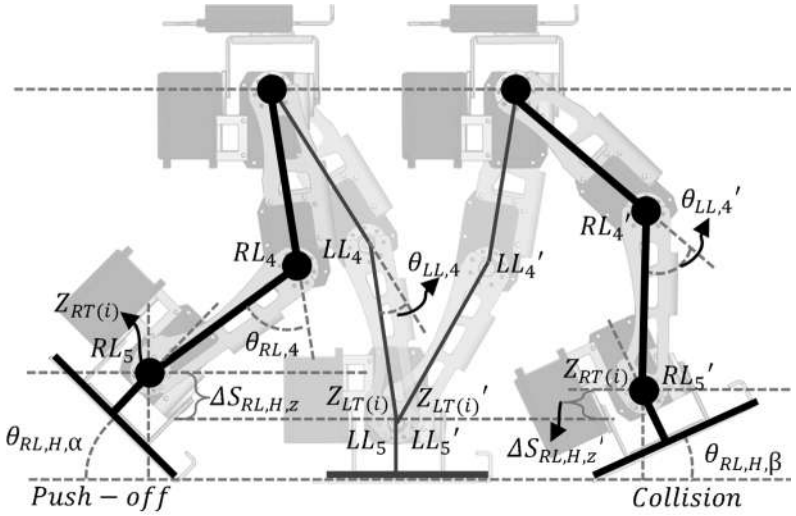

Figure 15. Heel-Strike gait model of EROS humanoid robot

Where $\theta_{RL,H,\alpha}$ and $\theta_{RL,H,\beta}$ is right ankle angle in RL_5 joint (ankle pitch angle) when push-off and collision state. $\theta_{LL,H,\alpha}$ and $\theta_{LL,H,\beta}$ is left ankle angle in LL_5 joint (ankle pitch angle) when push-off and collision state. $\theta_{RL,4}$ and $\theta_{LL,4}$ is knee joint angle in right and left legs when push-off state. $\theta_{RL,4}'$ and $\theta_{LL,4}'$ is knee joint angle in right and left legs when collision state. $\Delta S_{RL,H,z}$ and $\Delta S_{LL,H,z}$ is difference distance between $Z_{RT,(i)}$ and $Z_{LT,(i)}$ when push-off state. $\Delta S_{RL,H,z}'$ and $\Delta S_{LL,H,z}'$ is difference distance between $Z_{RT,(i)}$ and $Z_{LT,(i)}$ when collision state. $C_{H,\alpha\beta e}$ is constant gain value to arrange gain in $\theta_{RL,H,\alpha}$ and $\theta_{RL,H,\beta}$.

C. Landing Optimization with Deceleration

In the landing process, the transition value to reach $t_{LT,(i)}$ has a constant value of T_{mn} (explained in walking trajectory system). In the walking trajectory system T_{mn} is always constant value. This value of T_{mn} is summed with last transition value ($T_{T(i)} = T_{T(i-1)} + T_{mn}$). Because of it, the speed of steps will comparable with value of T_{mn}. When transition of each step at high speed, it will cause a huge reaction moment. With a huge reaction moment, the stability of robot when walking is disrupted even fall. Therefore, the deceleration speed of steps in landing process is done by using arrangement value of T_{mn}.

Figure 16 is explaining about dividing the area of $T_{LT,n}$ (left leg) in SSP to DSP state. The area of left leg is similar with right leg ($T_{(RT,LT),n} = (\ T_{LT,n} = T_{RT,n})$) These areas have a number depend on speed of steps and detail deceleration needed, so that when deceleration process the robot still in stable condition. To arrange the deceleration process, the value of T_{mn} is arrange with equation bellow.

$$T_{(RT,LT),n} = {}^{\Delta S_{(RT,LT),x,(i)}}/_{n} \tag{48}$$

$$T_{mn,DC(n)} = \log\left(T_{mn,DC(n-1)} - C_{DC,e}\right) \times T_{mn} \quad (49)$$

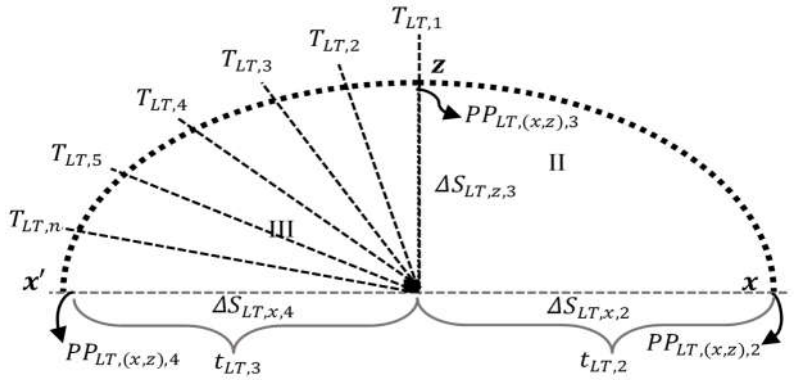

Figure 16. Dividing the area of $T_{LT,n}$ in SSP to DSP state (landing process)

Where $T_{(RT,LT),n}$ is number of areas in left (LT) and right (RT) leg after divided. $\Delta S_{(RT,LT),x,(i)}$ is difference between present and future point reference in X axis from trajectory system. T_{mn} is difference value between transition process in each step. $T_{mn,DC(n)}$ is present difference value between transition process in each step after deceleration process. $T_{mn,DC(n-1)}$ is last difference value between transition process in each step after deceleration process. $C_{DC,e}$ is constant gain value to arrange gain in $T_{mn,DC(n)}$. The result after deceleration process can be seen in Figure 17.

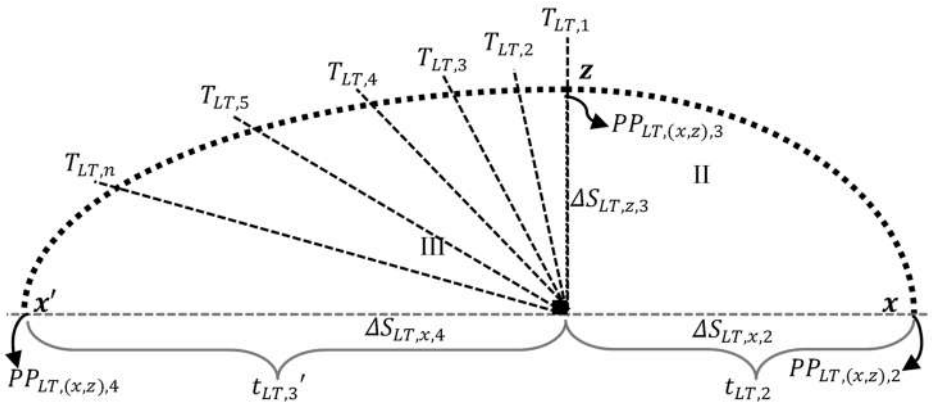

Figure 17. Dividing the area of $T_{LT,n}$ in SSP to DSP state (landing process)

By using LOG function, the deceleration will not drop in linier condition. Because of that, the robot will be too much to lose the speed during walking. With adding this algorithm, the stability of robot is maintained.

D. Comparison of Implementation (before and after optimization)

As mentioned in the originality section, this paper only discus about the problem caused by synthetic grass filed without a control system. In previous research of EROS, the walking trajectory system is used in the flat field (green

carpet). But in the new competition, the field was changed with a synthetic grass field. Because of it, when EROS is using previous walking trajectory system, the EROS always falling (the foot always stuck inside the synthetic grass surface). The idea to solve this problem is make the EROS robot walk like human. By using EROS system, the previous walking trajectory system is added with landing optimization with deceleration process and heel-strikes gait optimization process during the process of Single Support Phase (SSP) to Double Support Phase (DSP). So, the EROS robot will walk look like more human. Landing optimization with deceleration is used to arrange the velocity of landing process by sending the transition time value ($T_{mn,DC(n)}$). This section is replacing the constant velocity in previous system (walking trajectory system). Heel-strikes gait optimization is replacing the angle in ankle joint (directly to the robot joint). Both of optimization is explaining more clearly in the Figure 18 and Figure 19.

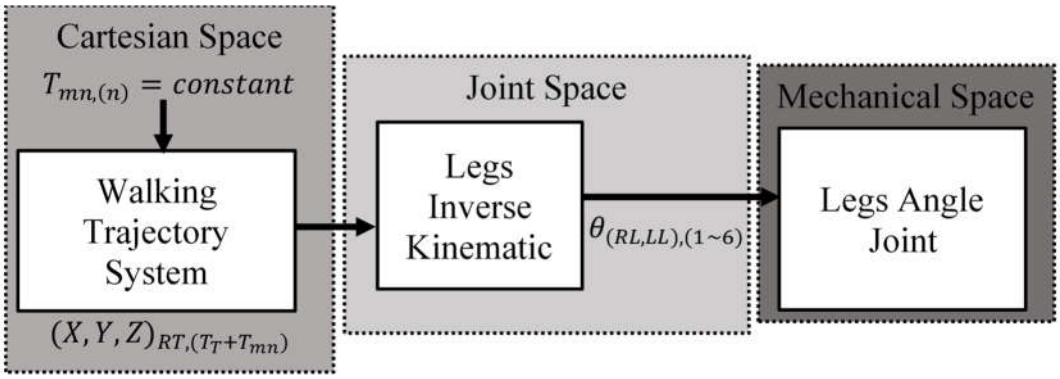

Figure 18. Walking locomotion previous version (flat field: green carpet)

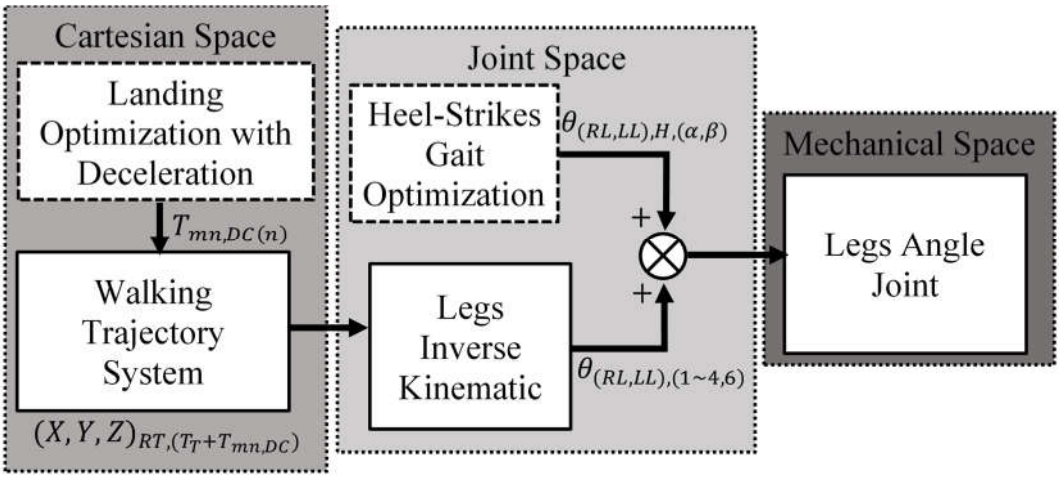

Figure 19. Walking locomotion newest version (synthetic grass field)

Based on Figure 17 and Figure 18, the mean of optimization process is how to make a better solution from desired system into a new complex system as mentioned in the originality section.

5. EXPERIMENT AND ANALYSIS

The experiment explains about robot stability during walking locomotion with optimization and during walking locomotion without optimization. The experiment is started with the difference of synthetic grass and several variations of walk distance in each step. The method of experiment is measuring angle body (Pitch) in *X* axis during walking motion. This experiment uses 4 kinds of different synthetic grass. The specification of the synthetic grass is shown in Figure 20, Figure 21, Figure 22 and Figure 23.

Figure 20. Pedestal 1 (synthetic grass with 50mm of grass height)

Figure 21. Pedestal 2 (synthetic grass with 30mm of grass height)

Figure 22. Pedestal 3 (synthetic grass with 10mm of grass height)

Figure 23. Pedestal 4 (green carpet with 4mm of thickness)

In Figure 20, the synthetic grass has a very rough and tender texture. The distance of each grass is 10mm. So, it is look rarely. The length of each grass is 50mm with irregular direction. In Figure 21, the synthetic grass has a very smooth and tender texture. The distance of each grass is 5mm. So, it is looks solid. The length of each grass is 30mm with same direction. In Figure 22, the synthetic grass has a very rough and tender texture. The distance of each grass is 5mm. So, it is look solid. The length of each grass is 10mm with same direction. In Figure 23, the carpet has a very smooth and soft texture with 4mm

of thickness. The experiment is explained with 3 represented result of experiment. Those experiment only using synthetic grass type 1 (pedestal 1) with different variation of walk distance in each step. The experiment has limitation distance about 6 meters of total distance.

The first experiment using walk distance about 8mm each step on the pedestal 1. The result can be seen in Figure 24. The Cartesian data input parameter for walking trajectory is X = 8mm, Y = 45mm, and Z = 100mm.

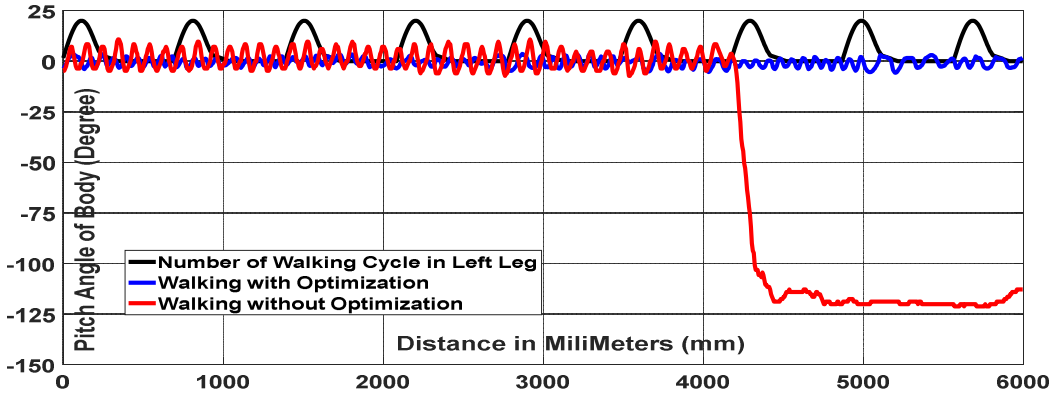

Figure 24. Experiment result using distance: 8mm each step of walk

In Figure 24, the red line is the stability graph during walking locomotion without optimization. It has average vibrations at ± 19^0. But at distance about 4460 mm or in step 6, the robot is falling. So, the walking locomotion is in unstable condition. The blue line is the stability graph during walking locomotion with optimization. It has average vibrations in ± 6^0. The robot does not fall. So, the walking locomotion is in stable condition. The walking locomotion without optimization have 4.46 meters of distance or 6 steps of walk and the walking locomotion with optimization have 6 meters of full distance or 9 steps of walk. The difference of distance in (%) is 25.67%. The implementation experiment in pedestal 1 is shown in Figure 25.

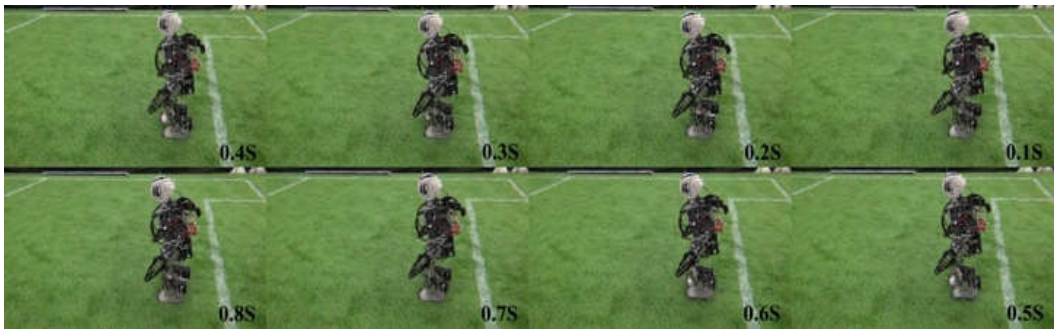

Figure 25. Implementation result using distance: 8mm each step of walk

The second experiment using walk distance about 16mm each step on the pedestal 1. The result can be seen in Figure 26. The Cartesian data input parameter for walking trajectory is X = 16mm, Y = 35mm, and Z = 95mm.

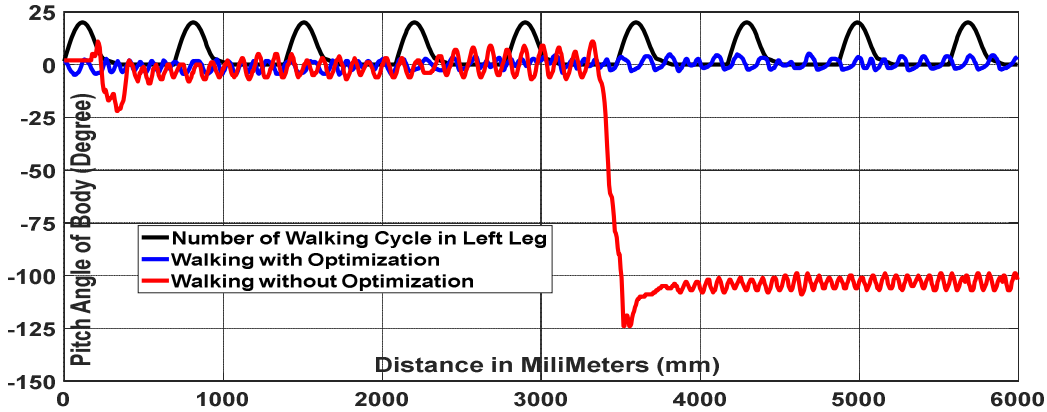

Figure 26. Experiment result using distance: 16mm each step of walk

In Figure 26, the red line is the stability graph during walking locomotion without optimization. It has average vibrations at ± 22^0. But at distance about 3570 mm or in step 5, the robot is falling. So, the walking locomotion is in unstable condition. The blue line is the stability graph during walking locomotion with optimization. It has average vibrations in ± 8^0. The robot does not fall. So, the walking locomotion is in stable condition. The walking locomotion without optimization have 3.57 meters of distance or 5 steps of walk and the walking locomotion with optimization have 6 meters of full distance or 9 steps of walk. The difference of distance in (%) is 40.5%. The implementation experiment in pedestal 1 is shown in Figure 27.

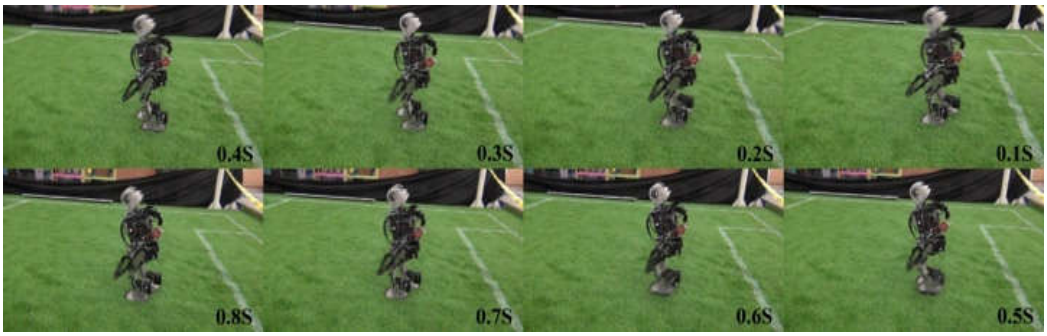

Figure 27. Implementation result using distance: 16mm each step of walk

The third experiment using walk distance about 24mm each step on the pedestal 1. The result can be seen in Figure 28. The Cartesian data input parameter for walking trajectory is X = 24mm, Y = 25mm, and Z = 90mm.

In Figure 28, the red line is the stability graph during walking locomotion without optimization. It has average vibrations at ± 21^0. But at distance about 2680 mm or in step 4, the robot is falling. So, the walking locomotion is in unstable condition. The blue line is the stability graph during walking locomotion with optimization. It has average vibrations in ± 7^0. The robot does not fall. So, the walking locomotion is in stable condition. The walking locomotion without optimization have 2.68 meters of distance or 4 steps of walk and the walking locomotion with optimization have 6 meters of full

distance or 9 steps of walk. The difference of distance in (%) is 55.33%. The implementation experiment in pedestal 1 is shown in Figure 29.

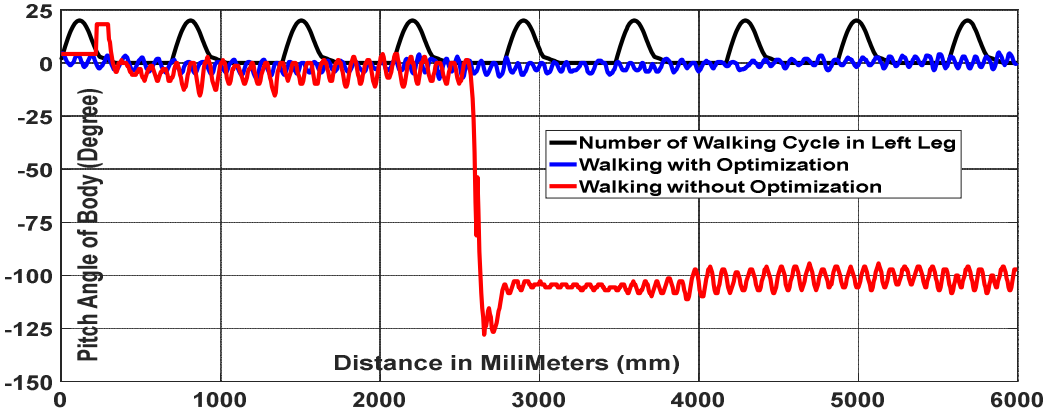

Figure 28. Experiment result using distance: 24mm each step of walk

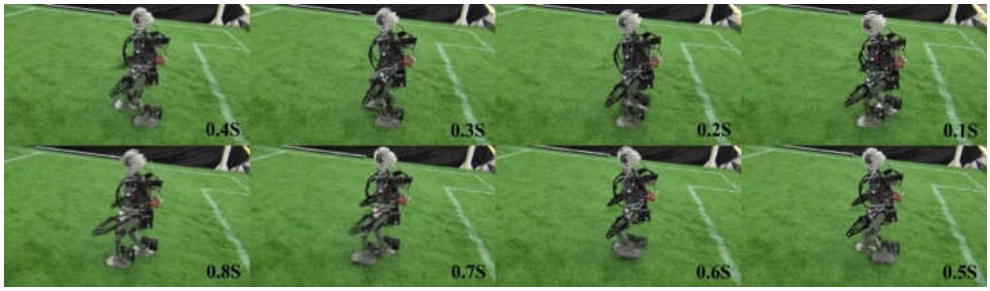

Figure 29. Implementation result using distance: 24mm each step of walk

More complete experiment result with variation of walk distance in each step and several types of synthetic grass is shown in Table 7 and Table 8. The system without optimization experiment is shown in Table 7.

Table 7. Experiment Result of Walking Locomotion without Optimization

FORWARD DIRECTION				
Distance (cm)	Pedestal 1	Pedestal 2	Pedestal 3	Pedestal 4
0	STABLE	STABLE	STABLE	STABLE
4	UNSTABLE	UNSTABLE	STABLE	STABLE
8	UNSTABLE	UNSTABLE	STABLE	STABLE
12	UNSTABLE	UNSTABLE	UNSTABLE	STABLE
16	UNSTABLE	UNSTABLE	UNSTABLE	STABLE
20	UNSTABLE	UNSTABLE	UNSTABLE	STABLE
24	UNSTABLE	UNSTABLE	UNSTABLE	UNSTABLE
BACKWARD DIRECTION				
Distance (cm)	Pedestal 1	Pedestal 2	Pedestal 3	Pedestal 4
8	STABLE	STABLE	STABLE	STABLE
16	STABLE	STABLE	STABLE	STABLE
24	UNSTABLE	UNSTABLE	UNSTABLE	STABLE

In this experiment result, the unstable condition is more dominant than stable condition. This condition happened because the robot is losing their

stability during the walking locomotion in the Single Support Phase (SSP) to Double Support Phase (DSP) or step 2 (landing). The toe of the foot was landed first, and it caused the toe stuck in the synthetic grass. The landing speed of steps are also makes toe stuck harder.

The system with optimization experiment is shown in Table 8. From information in Table 5, the stable condition is more dominate than unstable condition. The robot walks more stable in this experiment because some factors that causes the robot in unstable condition at the experiment without optimization is minimized by using optimization model (heel strikes gait and landing deceleration). When the walking trajectory in the Single Support Phase (SSP) to Double Support Phase (DSP) or step 2 (landing). The toe of the foot will be landed at the last time and the toes will not stuck in the syntetic grass because the implementation of heel-strikes gait optimization. The foot was landed slowly and softly because the landing speed of steps has been decelerated by using landing optimization with deceleration. It is make the toes will not stuck harder than before.

Table 8. Experiment Result of Walking Trajectory without Optimization

FORWARD DIRECTION				
Distance (cm)	Pedestal 1	Pedestal 2	Pedestal 3	Pedestal 4
0	STABLE	STABLE	STABLE	STABLE
4	STABLE	STABLE	STABLE	STABLE
8	STABLE	STABLE	STABLE	STABLE
12	STABLE	STABLE	STABLE	STABLE
16	STABLE	STABLE	STABLE	STABLE
20	UNSTABLE	UNSTABLE	STABLE	STABLE
24	UNSTABLE	UNSTABLE	UNSTABLE	STABLE
BACKWARD DIRECTION				
Distance (cm)	Pedestal 1	Pedestal 2	Pedestal 3	Pedestal 4
8	STABLE	STABLE	STABLE	STABLE
16	STABLE	STABLE	STABLE	STABLE
24	UNSTABLE	UNSTABLE	UNSTABLE	STABLE

In the experiment without optimization, the comparison of stability in per one hundred percent (%) is when on pedestal 1 the percentage of stability is 10%, when on pedestal 2 the percentage of stability is 10%, when on pedestal 3 the percentage of stability is 30%, and when on pedestal 4 the percentage of stability is 80%.

In the experiment with optimization, the comparison of stability in per one hundred percent (%) is when on pedestal 1 the percentage of stability is 70%, when on pedestal 2 the percentage of stability is 70%, when on pedestal 3 the percentage of stability is 80%, and when on pedestal 4 the percentage of stability is 100%.

In the all experiment of stability, the efficiency is obtained by averaging the stability value. Efficiency walking trajectory without optimization is 32% and efficiency walking trajectory with optimization is 80%. The experiment is using 4 types of pedestal, so it can be knowing the stability characteristic of

the robot. The main result of this experiment is the stability while using optimization is much better than without the optimization. The results can be seen from efficiency comparison of stability.

6. CONCLUSION

In this paper, walking trajectory model with landing optimization with deceleration and heel strikes gait optimization has been applied into EROS humanoid robot. The combination of both optimization systems has been tested. By adding landing optimization with deceleration and heel-strikes gait optimization into walking trajectory system for EROS humanoid robot, the stability of walking was increased from 32% to 80%.

To reach 90% - 100% of stability, control system will be added into EROS humanoid robot in the future work.

Acknowledgements

This document is dedicated to Electronics Engineering Polytechnics Institute of Surabaya (EEPIS), especially to department of electronics engineering and EEPIS Robotics Research Center (ER2C).

REFERENCES

[1] RoboCup Federation, **RoboCup Soccer Humanoid League Rules and Setup**, pp. 1–2, 2016.

[2] R. Dimas Pristovani, W. M. Rindo, B. Eko Henfri, Kh. Achmad S, and Pramadihanto. D, **Basic walking trajectory analysis in FLoW ROBOT**, In*: 2016 International Electronics Symposium (IES)*, pp 333–338, 2016.

[3] S. Kagami, K. Nishiwaki, J. J. Kuffner Jr, Y. Kuniyoshi, M. Inaba, and H. Inoue, **Online 3D Vision, Motion Planning and Biped Locomotion Control Coupling System of Humanoid of Humanoid Robot: H7**, in *Proc. Of the 2002 IEEE/RSJ Intl. Conference on Intelligent Robots and System EPFL*, pp. 2557–2563, 2002.

[4] Steven H. Collins, Martijn Wisse, Andy Ruina, **A Three-Dimensional Passive-Dynamic Walking Robot with Two Legs and Knees**, in *The Int. Journal of Robotics Research,* Vol. 20, No. 7, pp. 607–615, 2001.

[5] Azhar Aulia S, R. Dimas Pristovani, Ach. Subhan Kh, **Implementasi Pembangkitan Pola Gerakan Berjalan Berbasis Trajectori Joint Pada Robot EROS-1**, in *Proc. Indonesian Symposium on Robot Soccer Competition*, pp. 45–48, 2013.

[6] Azhar A.S, Ach Subhan Kh, Indra Adji. S, **Acceleration and deceleration optimization using inverted pendulum model on humanoid robot eros-2**, in *Proc. Indonesian Symposium on Robot Soccer Competition 2*, pp. 17–22, 2014.

[7] R. Dimas Pristovani, Ach Subhan Kh, **Deceleration Optimization Landing For Trajectory Walking On Robot Eros 5 (Eepis Robosoccer 5)**, in *Proc. Indonesian Symposium Robot Soccer Competition 3,* pp. 24–30, 2015.

[8] Reza N. Jazar, ***Theory of Applied Robotics: Kinematics, Dynamics, and Control***. *Springer Science + Business Media LLC,* pp. 199–296, 2007.

[9] Ach Subhan Kh, Iwan Kurnianto W, R. Dimas Pristovani, **EROS TEAM – Team Description for Humanoid KidSize League of RoboCup 2015**. In *19th RoboCup International Symposium-Team Description Papers,* pp. 1–7, 2015.

[10] R. Dimas Pristovani, D. Raden Sanggar, and Pramadihanto. Dadet, **Implementation of Push Recovery Strategy using Triple Linear Inverted Pendulum Model in "T-FloW" Humanoid Robot**, in *IOP Journal of Physics: Conference Series,* vol. 1007, pp. 1–13, 2018.

[11] R. Dimas Pristovani, B. Eko Henfri, D. Raden Sanggar, and Pramadihanto. Dadet, **Walking Strategy Model based on Zero Moment Point with Single Inverted Pendulum Approach in "T-FLoW" Humanoid Robot,** in *2017 2nd International Conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE),* pp. 217–222, 2017.