

Slope Balancing Strategy for Bipedal Robot Walking Based on Inclination Estimation Using Sensors Fusion

Riyanto, Carmadi Machbub, Hilwadi Hindersah, and Widyawardana Adiprawita

School of Electrical Engineering and Informatics
Institut Teknologi Bandung
riyanto@bppt.go.id

Abstract: The ability of a biped robot to run on various types of floors and slopes can be an additional guarantee that the robot is capable of performing various tasks as can be done by humans. This paper is the result of a study on a bipedal robot that only consists of a body and two legs with two feet to use the ankle-hip and time delay strategy to anticipate the effects of the slope of the floor during walking. The result of the research is the robot could adapt the change of slope up to 12 degrees while walking with so that it can keep the balance and does not fall. The hip and ankle strategies are applied together on the inclined floor by dividing the slope impact on the hip and ankle joint to a certain ratio, while the time delay strategy is done when the robot walks on a sloping surface to increase the binding power of the foot to the floor.

Keywords: Biped Robot, Posture Controller, Inclination Estimation, Balance Strategy, Sensors Fusion, Arduino

1. Introduction

Adaptation to various floor slopes is one of the capabilities that must be possessed by a biped robot to keep the robot able to walk without falling. Many methods are carried out by researchers to make the robot able to adapt to the slope of the floor, including what is done by authors at [1] by reading the slope of the floor before starting the robot walking motion then use the floor slope reading as a reference to compensate the gravity for the Bioloid Premium robot, or using the error data from the gyroscope reading which is used as input control with the PD method for strategy ankle on a Bioloid GP robot as done by Martin et al. [10].

Posture control can be used to ease the work of the servo, especially when the robot gets external disturbance in the form of tilt, push or pull. Posture control is a control method inspired by humans when facing the disturbances, in the form of adjusting the shape of the body's posture to compensate for the disturbance. Ankle and hip strategies are the strategies that are used by most of biped robot researcher when the robot have to do adaptation to the unbalance condition. The ankle and hip strategies are carried out alternately and simultaneously according to the needs and desires of the researchers. Ankle strategy uses the ankle torque to recover from the external push, was done by Huang [5] that used ankle torque compensation to reject a small disturbance force in which ZMP fluctuation was compensated during walking. Vadakkepat et. al. [2] used online ZMP compensation to reject an external disturbance using ankle torque. Hip strategy controls the hip torque against the external force. Horak et. al. At [3] provided a mainframe structure regarding the hip strategy. Nenchev et. al.[7] proposed a combination of ankle and hip strategies to achieve balance recovery from an impact force with a simple planar biped model and the reaction null space method. The ankle and hip strategies also are used by Guihard et al.[8] to adapt the robot's balance to interference from outside of the robot in the form of encouragement and attraction. The ankle, hip and stepping strategies carried out with the help of predictive control are carried out by Aftab et al. [9] to adjust the step duration. Hemami et. al. [4] made the use of a feedback linearization technique using hip and ankle torque against an external disturbance force. Kim at al. in [6] employed and elaborated on ZMP based and inclination based sensory reflex control to stabilize standing motion and biped walking using IMU and FSR. The robot is able to walk on the floor with 7 degrees inclination.

Received: March 15th, 2019. Accepted: September 20th, 2019

DOI: 10.15676/ijeel.2019.11.3.6

In this paper, the authors propose a strategy to deal with the slope of the floor using the strategy of hip, ankle, and time delay. This strategy is applied to the walking motion generator developed previously by authors in [1], using the results of the floor slope detection to determine the size of the hip, ankle, and time delay strategies that will be applied to biped robot.

This paper is made with the following arrangement. The robot structure used in the experiment will be described in Chapter 2 of this paper. In Chapter 3, briefly explained the walking motion generator adopted from [1] and important poses that are used as key poses for the application of the strategies to be applied. This chapter also describes the development of gravity compensation for improving gait for the biped robot. In Chapter 4, the author describes the method of detecting the slope of the floor which is an important element before applying the balancing strategy to the robot. In Chapter 5, we will explain the method of hip, ankle, and delay used in the robot to become a strategy applied to maintain balance gait on flat or sloping floors. Experimental results conducted on robots are presented in Chapter 6, conclusions will be presented in Chapter 7.

2. Robot Structure

The robot used in the experiment is a modification of the Type A Bioloid Premium robot from Robotis, Korea. The robot adopts the hip and legs of premium type A Bioloid robots, but by replacing the CM-530 controller with Arduino. Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software [12]. The use of Arduino as a substitute for CM-530 due to the limitations of the I/O port on the CM-530 which only has 8 ports, so it is not able to meet the needs of I/O ports which will be used as inputs from sensors and remote controls that require more I/O port. The robot then is called Ardubio-Walker.

Ardubio-Walker is a biped robot consisting of a body and two legs made by mimicking the structure of human legs. Ardubio-Walker is composed of total 12 degrees of freedom (DoF) consisting of 6 DoF on each leg with a 2 DoF on each ankle, 1 DoF on each knee, and 3 DoF on each hip. The Ardubio-Walker structure, consisting of links and 12 Dynamixel AX-12A servo motors from Robotis, and it is equipped with an IMU sensor with type MPU 6050 that is attached to the robot's pelvis. Ardubio-Walker controls are carried out using the Arduino module to read the sensor data, to process sensor and desired data, and to provide control signals on Dynamixel motors. The developed Ardubio-Walker is shown in Figure 1 which is displayed in pose 1 (standing still pose) from the sagittal and frontal view.

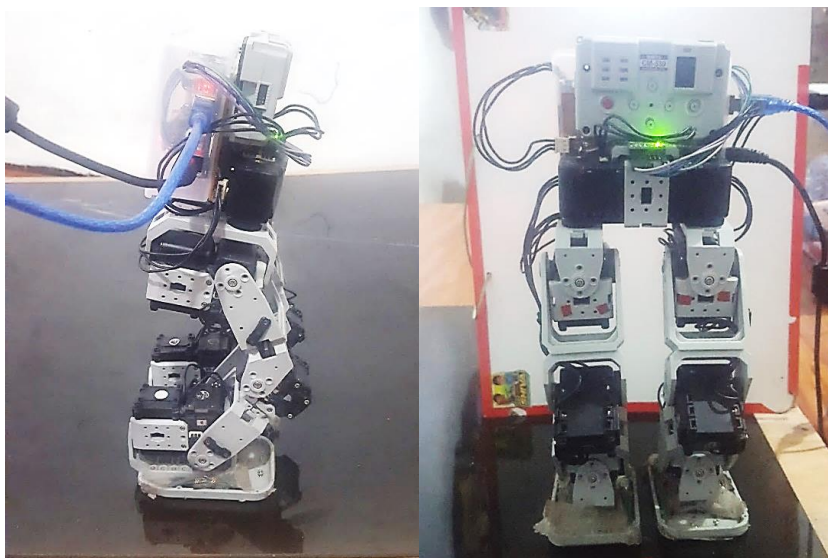


Figure 1. The developed Ardubio-Walker

The physical parameter of Biped Walker is shown in Figure 2 and Table 1. Figure 2 shows Ardubio-Walker as a sticky diagram with a relative angle θ_i of the i -th link to the $(i+1)$ -th link from the sagittal and frontal view. Table 1 shows the length of the link of the robot in millimeters.

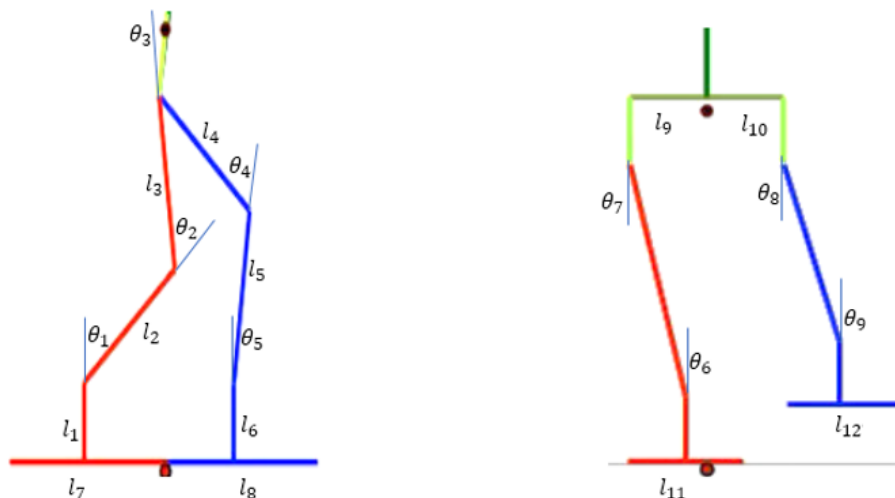


Figure 2. Ardubio-Walker Links and Angle Positions

The black dot on the pelvic part of the robot is assumed to be the position of the center of mass (CoM) of the Ardubio-Walker. The red dots on the floor are projections of the CoM of the robot.

Table 1. Ardubio-Walker Links Length

Link Name	Length (mm)	Link Name	Length (mm)
l_1	35	l_7	100
l_2	75	l_8	100
l_3	75	l_9	40
l_4	75	l_{10}	40
l_5	75	l_{11}	60
l_6	35	l_{12}	60

Dynamixel servos that are used as Ardubio-Walker joints are produced by Robotis, a Korean company that also makes the Bioloid robot kits. The AX-12A robot servo has 55 g weight and 1/254 gear reduction ratio as shown in Figure 3 has the ability to track its speed, temperature, shaft position, voltage, and load. All of the sensor management and position control of AX-12A is handled by the servo's built-in microcontroller [11].



Figure 3. Dynamixel AX-12A

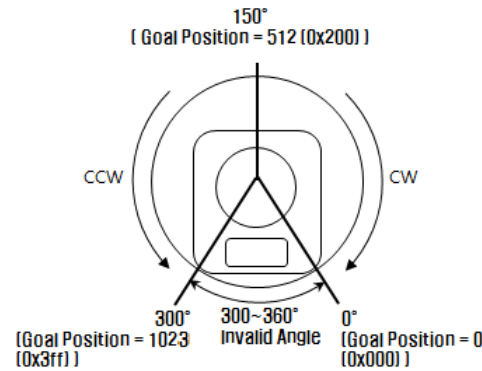


Figure 4. Current position value of Dynamixel.

The Dynamixel value is an integer value from 0 to 1023, while the angle degree value is a real value from 0 to 300 degrees as shown in Figure 4, which then for the convenience of computing is converted to a value from -150 to 150 degrees. Each one of the Dynamixel values is equal to 0.29 degrees of angle value based on the conversion equation of 1023 Dynamixel values which is equivalent to 300 degrees of angle value. Conversion of the angle value to Dynamixel value through rounding poses which causes a slight inaccuracy in the conversion value of Dynamixel. One degree of angle value should have a Dynamixel value of 3.41 because it is rounded up to have a value of 3 which is equivalent to an angle value of 0.88 degrees, which means there is a difference in value of 0.12 degrees.

In the Ardubio-Walker consisting of 12 joints, the difference in the conversion of the angular values to each Dynamixel value used by robots can be a challenge in achieving robotic balance.

3. Walking Pattern Generation

The walking pattern generator that is used in this paper adapts the walking pattern generator built by [1], which produces data-driven on sticky 5 link robot models as in Figure 2, in the form of Dynamixel 7 to 18 (12 DoF) values on the Ardubio-Walker along with speed based on hip movements obtained from linear inverted pendulum movement patterns, and swing foot movements obtained from the cycloid motion. Relative angles R of the robot link is obtained with the inverse kinematics, i.e.

$$R = [\theta_1^d, \theta_2^d, \theta_3^d \dots \theta_9^d] \tag{1}$$

The relative angle of the link is then converted to the value of Dynamixel so that the Dynamixel value D is obtained as data-driven for Arduino control board.

$$D = [d_7^d, d_8^d, d_9^d, \dots d_{18}^d] \tag{2}$$

along with the speed matrix \dot{D} for each joint

$$\dot{D} = [\dot{d}_7^d, \dot{d}_8^d, \dot{d}_9^d, \dots \dot{d}_{18}^d] \tag{3}$$

The speed of each joint is calculated from the position of each joint concerning the previous join position in every sampling time t_s . The detail of the conversion function can be seen in [1].

A. Center of Mass

The reference for the center of mass trajectory uses a simple sinusoidal motion with equation as follow

$$CoM(t) = -A \cos \omega(t-t_0) \quad (t \geq Tstep, t_0=Tstep) \tag{4}$$

with ω is the frequency of the function that describes the walking trajectory of the hip, $Tstep$ is the time duration for one step and A denotes the transverse swing amplitude. The negative sign indicates that the right leg as the first leg to swing. Ardubio-Walker always starts the walking process with the right leg. Equation (4) is the basis for forming a trajectory equation for the pelvis which becomes the key joint on the robot movement.

The center of mass of Ardubio-Walker is assumed to be located on the pelvis of the robot. The coordinate of the pelvis is calculated with the Denavit Hartenberg (DH) parameter method with the reference to the stance foot of the Ardubio-Walker. DH parameter for the right foot and left foot can be seen in Figure 5 and Table 2.

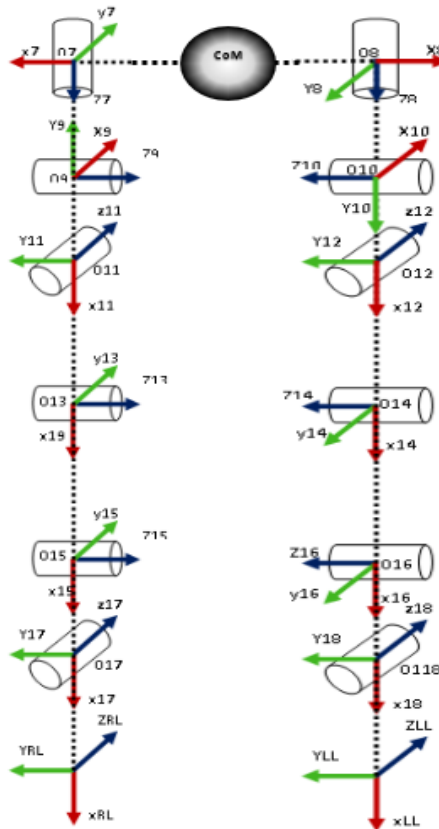


Figure 5. DH Parameter of Ardubio-Walker

Table 2. DH parameter of Ardubio-Walker for the right foot and left foot as the stance foot

θ_i	d_i	a_i	α_i
0	0	$L0$	0
θ_{17}	0	0	-90
θ_{15}	0	$L1$	0
θ_{13}	0	$L2$	0
θ_{11}	0	0	90
θ_9	0	$L3$	-90
0	$L4$	0	90

(a) Right Foot

θ_i	d_i	a_i	α_i
0	0	$L0$	0
θ_{18}	0	0	-90
θ_{16}	0	$L1$	0
θ_{14}	0	$L2$	0
θ_{12}	0	0	90
θ_{10}	0	$L3$	-90
0	$L4$	0	90

(b) Left Foot

so that the CoM position coordinate (c_x, c_y, c_z) is obtained with reference to the stance foot is the left foot with the following equations (5).

$$c_x = -L_4 C_{17} C_{15.13.11} S_9 + L_4 C_{17} C_9 + L_3 S_{17} C_{15.13.11} C_9 + L_3 C_{17} S_9 + L_2 S_{17} S_{15.13} + L_1 S_{17} C_{15}$$

$$c_y = -L_4 C_{17} C_{15.13.11} S_9 + L_4 C_{17} C_9 + L_3 S_{17} C_{15.13.11} C_9 + L_3 C_{17} S_9 + L_2 S_{17} C_{15.13} + L_1 S_{17} C_{15}$$

$$c_z = L_4 C_{17} C_{15.13.11} S_9 - L_4 S_{17} C_9 + L_3 C_{17} C_{15.13.11} C_9 - L_3 S_{17} S_9 + L_2 C_{17} C_{15.13} + L_1 C_{17} C_{15} + L_0 \quad (5)$$

with $L_0 = 35$ mm, $L_1 = 75$ mm, $L_2 = 75$ mm, $L_3 = 35$ mm, and $L_4 = 40$ mm. C_i and S_i are notations for $\cos(i)$ and $\sin(i)$, while $C_{i,j}$ and $S_{i,j}$ are notations for $\cos(i+j)$ and $\sin(i+j)$. Whereas if the reference foot is the right foot, then the angles with an even number which correspond to the angular numbers in the equation above are used.

The calculation of the center of mass with the DH method has limitations when the robot is on a sloping floor because it refers to the soles of the feet which causes inaccuracies in the position of the center of mass when the soles of the feet are on the slope of the floor. It is necessary to adjust the calculation of equation (5) by paying attention to the slope of the floor so that the assumption of the position measurement of the center of the robot's mass is more accurate.

B. Swing Foot Trajectory

Swing leg trajectory can be formed by the rolling circle function, which turns along the x -axis as shown in Figure 6. By looking at a single point on the edge of the circle, indicates the position of the current point when rolling. The z -axis indicates the position of the height and the x -axis is the direction of movement of the leg.

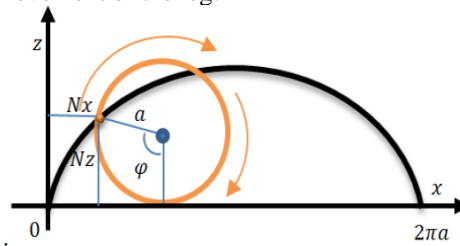


Figure 6. Rolling Circle Motion Generator

The equation for the swing foot is formed with the equation (6) and (7) as follow

$$Nxr(t) = a((\varphi(t) - \sin(\varphi(t)))) \text{ for } 0 < \varphi(t) < 2\pi \quad (6)$$

$$Nzr(t) = \delta a(1 - \cos(\varphi(t))) \text{ for } 0 < \varphi(t) < 2\pi \quad (7)$$

with $\varphi(t) = 2\pi t/T_{ssd}$ and $T_{ssd} = 2T_s$ is time for the single support phase and is δ a multiplier to get the expected step height. The left foot uses the same trajectory equation as the right foot but with the time lag T_s . Equation (6) and (7) is the basis of summoning the coordinates of the left and right ankle coordinates which are another key joints of the robot leg movement.

The swing foot in the walking pattern generation is planned to always level with the ground during leaving and striking the ground.

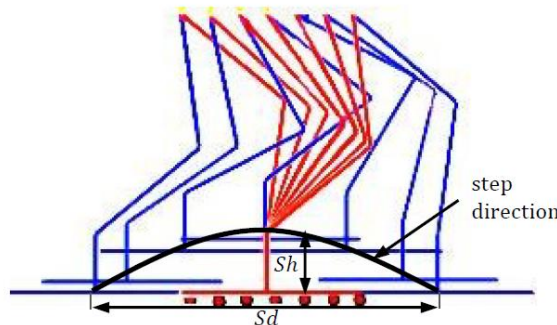


Figure 7. The Swing Foot Trajectory

Figure 7. shows the trajectory of the swing foot, with S_h is the step height, and S_d is the step distance.

C. Gravity Compensation

Gravity compensation is added to correct the final posture of the robot, especially in the phase of the swinging foot that is affected by gravity so that the posture that appears on the robot is different from the planned posture. The image of the planned posture and the postures affected by gravity can be seen in Figure 8.

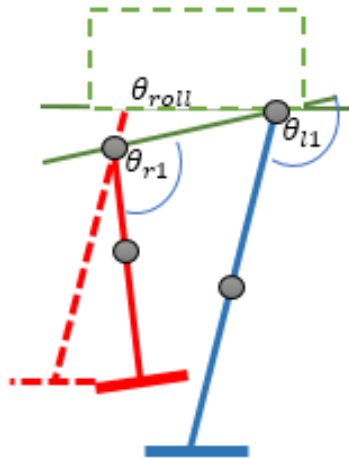


Figure 8. Planned posture and real gravity affected posture

Gravitational compensation given to the robot is developed from gravity compensation performed by [1] which only corrects the swinging leg. In this study, corrections were also made to the stance foot to maintain the slope of the upper body of the robot which is the same value as the correction of the swinging leg.

D. Joint Velocity

The speed of each joint is determined through a walking motion generator which is a function of time from the initial angle and the final angle of each joint in each state. Joint speed is obtained through the following equation

$$\dot{q}_i = \frac{q_i - q_{(i-1)}}{t_i} \tag{8}$$

where q_i is the destination angle value and $q_{(i-1)}$ is the initial angle value of each Dynamixel. The value of each Dynamixel is read in every state from the sensor built-in the Dynamixel

E. WPG State

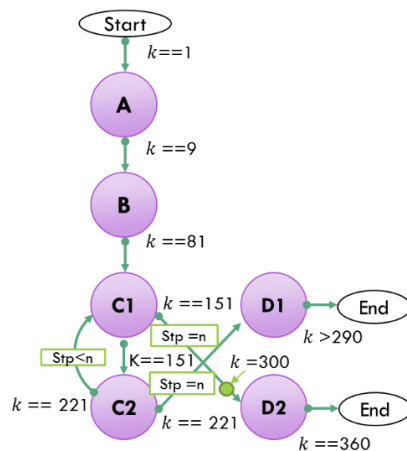


Figure 9. WPG State

We propose the walking state of WPG of Ardubio-Walker. WPG state consisting one state of the initial process (A), one state of the first step (B), two states of full step (C1, C2), and two states of final step (D1, D2), that depends on which foot that is stepping at last full step. The rule of the WPG state change is shown in Figure 9. Each state is carried out during counter k .

4. Inclination Detection

The tilt detection process in the initial phase is needed to make the robot able to adjust its initial posture quickly to adapt to the slope of the floor, to avoid the robot falling when it is placed. In the walking phase, tilt detection can provide correction values quickly and precisely on the hip and ankle to position the robot's center of mass to return to the support polygon. The measurement of the slope of the floor uses the IMU MPU 6050 sensor reading with 6 axes filtered with Kalman Filter to reduce or eliminate the noise on the measurement results due to sensor sensitivity. The inclination detection process is carried out on 2 robot conditions, namely in the initial condition (pose 0 or pose 1 robot), and in the transition condition (when there is an inclination change on the floor)

A. Kalman Filtering

An inherent problem with gyroscopes is the tendency for their output to drift over time. This drift occurs from noise and error compounding in the calculations over the time of operation, also known as integration error. One possible way to compensate for gyro drift is by using it in conjunction with an accelerometer. An accelerometer is used to measure acceleration within a certain range of motion. Movement is calculated similar to the gyro by using changes in the accelerometer's output voltage to determine changes in orientation. Depending on the angle of the device in relation to the ground, a microscopic amount of mass moves within the accelerometer circuit to cause this change in signal. An accelerometer can measure an absolute orientation. When a gyroscope and accelerometer are combined, it is possible to more accurately determine/measure absolute/relative orientation/movement since the devices are used to complement each other. But the use of gyro and accelerometer at IMU still produces data that is too noisy and drift, so a filter is needed to produce cleaner and more accurate data. In the filter process, noise from the accelerometer will be minimized, even eliminated.

The solution to the problem of linear filters in discrete data is explained by R. E Kalman in [13]. Kalman Filter is used to eliminate noise measurement from the IMU sensor. The disadvantage of this filter itself is that there is no standard in the method of getting the mathematical equations that will be used on the Kalman Filter [14].

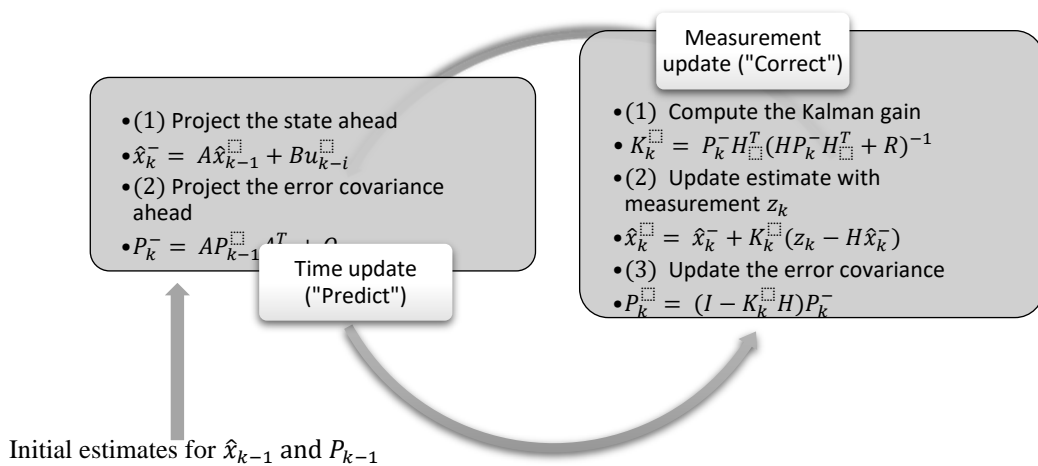


Figure 10. Kalman Filter Algorithm Scheme

First of all the filter will estimate the state of the system at a time, then get a measurement that contains noise to be used as feedback. After that, the equation is divided into two, namely the update time and measurement update. update times can also be called prediction equations and measurement updates called correction equations. The Kalman filter scheme and algorithm can be seen in Figure 10.

The Kalman filter modelling steps carried out in this study are as follows.

The state of the system at time k if given by:

$$\hat{x}_k = \mathbf{A}\hat{x}_{k-1} + \mathbf{B}u_k \quad (9)$$

where \hat{x}_k is the state matrix of the system at time k which is given by:

$$\hat{x}_k = \begin{bmatrix} \theta \\ \dot{\theta}_b \end{bmatrix}_k \quad (10)$$

with θ is the output of the accelerometer and $\dot{\theta}_b$ is the amount of bias from the gyroscope. To get the right angle, the bias value must be removed so that it is selected.

State transition model A which is applied to the previous state \hat{x}_{k-1} is defined as:

$$A = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \quad (11)$$

the control input model B , is defined as:

$$B = \begin{bmatrix} \Delta t \\ 0 \end{bmatrix} \quad (12)$$

with Δt is sampling time. The next is the control input u_k , in this case it is the gyroscope measurement in degrees per second ($^\circ/s$) at time k , this is also called the rate $\dot{\theta}$.

$$u_k = \dot{\theta}_k \quad (13)$$

the state equation is can be rewritten as:

$$\hat{x}_k = \mathbf{A}\hat{x}_{k-1} + \mathbf{B}\dot{\theta}_k \quad (14)$$

since the bias can not calculated directly based on the rate, the bottom of the matrix is set to zero.

\mathbf{Q}_k is the process noise covariance matrix and in this case the covariance matrix of the state estimate of the accelerometer and bias. The estimate of the bias and the accelerometer is assumed to be independent, so the value is equal to the variance of the estimate of the accelerometer and bias.

$$\mathbf{Q} = \begin{bmatrix} Q_\theta & 0 \\ 0 & Q_{\dot{\theta}_b} \end{bmatrix} \Delta_t \quad (15)$$

and measurement covariance matrix R is

$$\mathbf{R} = \text{var}(v_k) \quad (16)$$

the measurement noise is assumed the same and does not depend on the time k , so the equation (16) can be changed into

$$\mathbf{R} = \text{var}(v_k) = \text{var}(v) \quad (17)$$

Setting a small Q means we are confident about the prediction model. The large R along with a small Q means the estimation will prioritize respecting the prediction rather than the measurements. The consequence is that in case an unexpected phenomenon perturbs the system, the estimation will slowly be modified, since the filter does not believe the measurements compared to the prediction. By decreasing the delay (or the inertia) of the estimation, we can set a smaller R , by the cost of increasing noise. The main issue on setting these matrices is that there is no technique to calculate their values. In addition, each of their elements can be set individually and it is not trivial to understand how the filter is affected [15]. The value of Q_θ , $Q_{\dot{\theta}_b}$, and $\text{var}(v)$ are set to 0.001, 0.003, and 0.03.

B. Initial Detection

The process of tilt detection in initial conditions is carried out in pose 0 robot (sitting) and in the pose of 1 robot (standing still). The detection method is done by reading the IMU value and the value of the robot's posture. The IMU output data is a pitch and roll robot value, and the Dynamixel sensor output data is a Dynamixel value that shows the current pose of the robot.

Determination of slope is done through weighting from the results of reading the two sensors. The slope reading results are used as the initial input data for the hip, ankle and delay strategies that will be applied to the robot.

Weighting methods are carried out using experimental data to characterize poses and slopes CoM in various inclined planes.

The results of the initial detection are used to form the robot pose towards pose 2 (ready to walk pose), so that the robot is able to start steps on the slope with a posture that is able to anticipate instability due to the slope of the floor. This strategy is divided into two parts, namely those starting from pose 0 robot, and which starts from pose 1 robot.

The flow of initial detection is as follows

1. Read the IMU value on the robot when the robot is in pose 0 or pose 1
2. Read the value of the robot's position
3. In the initial detection process starting from pose 1, the ankle and hip compensation values are carried out which are already working on the robot's posture
4. Weighting is done on pitch value, roll value, CoM value, and posture value
5. The slope of the floor is currently legible
6. The compensation value of hip and ankle is fed to the robot according to the slope value
7. Robot starts pose 2
8. Start the walking process.

C. Transition Detection

When there is a change in the slope of the floor, the robot will detect the magnitude of the tilt change to change the magnitude of the ankle and hip correction to adjust the condition of the robot to tilt changes. The detection process is carried out by reading the IMU value, posture value, and ongoing ankle and hip correction values with the same weighting method as the initial detection. The value of the new slope reading will be added to the ongoing slope value.

The process of estimating the slope of the floor during walking is a challenge because pitch movements are deliberately made to swing back and forth during walking, so that when the right foot swings and when the left foot swings each has a different pitch deviation reference. The estimation process begins by determining the normal state of the robot when walking on a flat floor. Data collection is done by making the robot run at a slope of 1 degree to 5 degrees with each experiment carried out 10 times. At the slope i degree, posture with offset obtained at $(i+1)$ degree is used, so that a sensor reading data set is obtained on the robot while walking without compensation on each slope.

From the results of data collection, the fact is that there is a deviation of pitch angle readings by the sensors fusion as much as 1 degree to 1.5 degrees at 1 degree inclination. Based on these results it can be concluded that the change in slope during the robot can be read by reading the pitch angle values on the sensors fusion.

From the data collection obtained the estimated equation of slope of the surface of the floor during running as follows

$$\Delta\alpha = k(\theta_p^{meas} - \theta_p^{ref-i}) \quad (18)$$

with $\Delta\alpha$ is the value of the slope of the floor surface. The value of k is in the range [1,1.3] based on the reading of the pitch change value at each change of 1 degree slope by also taking into account the pitch deviation due to the acceleration of CoM. θ_p^{ref-i} is the pitch angle reference, and superscript i to mark the swing phase of the right foot or left foot So the new floor slope value is as follows

$$\alpha' = \alpha + \Delta\alpha \quad (19)$$

where α is the reading of the slope of the previous floor, and α' 'is an estimate of the slope of the new floor.

5. Walking and Balance Strategy

Walking is a loop between the single support phase (SSP) and the double support phase (DSP) which always passes through the touch phase to the floor. The pattern and speed of walking can be regulated by dividing the percentage between SSP and DSP. In normal human walking, the percentage between SSP and DSP is around 80% and 20% [8]. The higher the walking speed, the smaller the percentage used in the double leg phase. On flat and even floor, the walking motion generator built by [1] uses a percentage of 80% for SSP and 20% for DSP with a knee angle set at 150 degrees. The authors use the walking pattern generator built in [1] with 170 degree or almost stretch knee angle, by making improvements to the gravity compensation on the robot, namely by making corrections to the robot roll angle which is also affected when the robot's feet and the use of a balance strategy that uses a combination of ankle strategy and hip strategy and is equipped with a delay strategy to compensate for the way biped robots pay attention to the slope of the floor.

In the current strategy, before starting the walking phase, the robot is in pose 0 (sitting pose) or pose 1 (standing still pose). The slope of the initial floor is determined by reading the IMU sensor value and the pose value on the robot. The results from the reading become the initial inclination which initial reference calls the initial compensation value on the hip and ankle in pose 2 (ready to walk pose).

The ankle strategy is a strategy that regulates the ankle angle to compensate for the interference with the robot. Force disorders can be either slope or push and pull that occurs on the robot. This strategy is widely used by researchers on biped robots to show the static balance of a robot in a moving field. When the ankle strategy is no longer able to compensate for interference with the robot, the researchers replace and/or combine with a hip strategy which is a strategy that regulates the slope of the robot's torso to position the center of mass of robot so as not to leave the robot support polygon.

To improve the balance of the robot, the authors added a delay strategy that regulates the walking speed of the robot when stepping on a sloping floor by adding time to the DSP. The results of strategy effectiveness are shown through experiments in the Chapter 6. Ankle strategy, hip strategy, and delay strategy use the data measurement of CoM from Dynamixel and inclination from IMU as the input of balance control.

A. Ankel and Hip Strategies

The ankle strategy is an ankle angle setting to return the mass center back to the support polygon when the robot's center mass projection begins to leave the robot support polygon due to the slope of the floor. While the hip strategy is a hip angle setting to return the center of mass back to the robot support polygon.

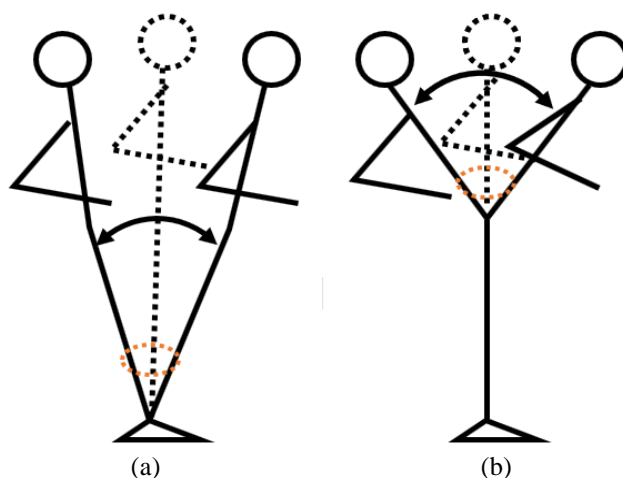


Figure 11. (a) Ankle Strategy and (b) Hip Strategy

The ankle strategy in the walker Bioloid is in the arrangement of joints with Dynamixel with number 15 and 16 for correction in the sagittal plane, and joint with Dynamixel with number 17 and 18 for correction in the frontal plane.

The ankle correction is done by the equations (20) and (21) as follows

$$d'_i = d_i + k_1\alpha \text{ for } i=15, 16 \tag{20}$$

$$d'_j = d_j + k_2\alpha \text{ for } j=17, 18 \tag{21}$$

with d is the Dynamixel value before the correction occurs, d' is the Dynamixel value after correction, and α is the slope of the floor from the reading on the Ardubio-Walker.

The hip strategy in Biolid Walker is a joint arrangement with numbers 11 and 12 for correction of the sagittal plane, and joints with numbers 9 and 10 for correction in the frontal plane. Correction of the hip is done with the following equation (22) and (23).

$$d'_k = d_i + k_3\alpha \text{ for } k=11, 12 \tag{22}$$

$$d'_l = d_j + k_4\alpha \text{ for } l=9, 10 \tag{23}$$

Determination of gain values k_1 , k_2 , k_3 and k_4 was carried out through a robot walking experiment on a flat surface with slopes of 0 to 12 degrees without any inclination transition in each experiment .

Compensation for the inclination of the floor in Ardubio-Walker is carried out with a strategy of dividing compensation. The division of compensation is done by dividing the ratio into 100% : 0%, 75% : 25%, 70% : 30%, 50% : 50%, 25% : 75%, and 0% : 100% for ankle strategies and hip strategies. Giving compensation of 1° slope is equivalent to the Dynamixel 4 value approach, so that for each addition of 1° slope, ankle and hip compensation is added to the planned ratio.

B. Delay Strategy

The delay strategy is used by the author to try to increase the binding power of the foot to the floor surface when the robot is walking on a sloping surface to reduce the risk of slipping. The delay strategy is in the form of adding double support phase (DSP) time to reduce shocks at the robot pitch angle so that the robot can maintain its balance when walking on a sloping floor. The added pause value is adjusted according to the slope of the floor. The greater the slope of the floor, the greater the pause given in the DSP phase. The additional delay used to adjust the speed of walking Ardubio-Walker uses the following equation

$$\Delta j' = \Delta j + k_5\alpha \tag{24}$$

with Δj is a delay time added to the change from state C1 to state C2 and vice versa, and the value of k_5 is obtained through experiments on varying slope of the floor.

C. System Architecture

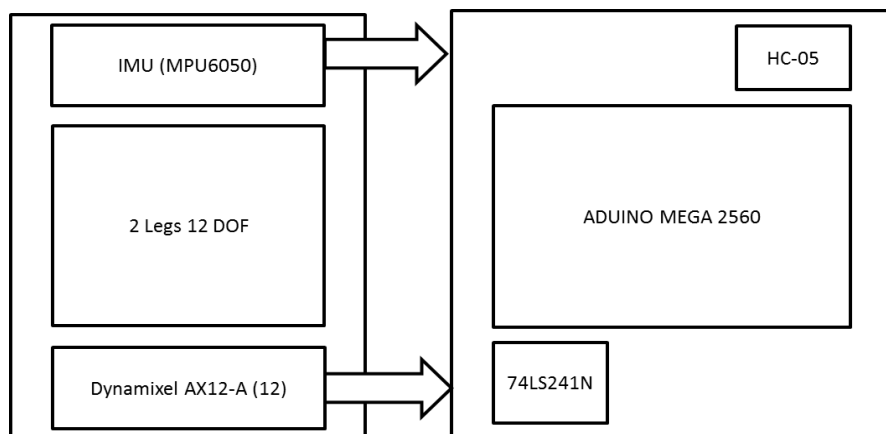


Figure 12 . System Architecture

The biped robot used is built with the system architecture as shown in Figure 12. The architecture of Ardubio-Walker consists of links and joints. The actuator as a joint consists of 12 Dynamixel AX12-A smart servo motors with independent microcontroller in each motor. Data communication from Dynamixel uses a 74S241N component with a half duplex system. The Arduino module uses Arduino Mega to read Dynamixel and IMU data as sensory data and drive Dynamixel as an actuator. To measure the tilt and the jerk on the robot, we used the IMU MPU 6050. To display output input data on a robot, serial communication and bluetooth are used.

The process of reading composite data to produce information on the slope of the ground is done by sensors fusion. By knowing the slope of the ground as a robot footing, the robot can provide ankle and hip feedback which is defined as FSA and FSH.

D. Hardware System

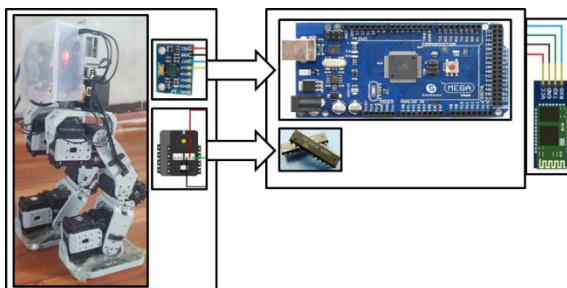


Figure 13. Hardware System of Ardubio-Walker

Hardware requirements on this system can be divided into four, namely sensory parts, actuators and robotic structures, processing, and data communication. The picture of each part can be seen in Figure 13.

- Sensory: there are two measurement devices namely IMU using MPU650, and the position of the joint angle using the Dynamixel position sensor. IMU consists of Gyro and Accelerometer. IMU is a component that measures Yaw, Pitch and Roll on a robotic hip.
- Links and Actuators: there are 12 Dynamixel that are used on systems connected with links that form a two-legged robot structure.
- Processing: Arduino Mega is used as a process of sensory data to obtain a control signal that can regulates the joint hip and ankle so that the expected CoM position correction is obtained.
- Data communication: There are two data communications in this system, namely communication of data from Dynamixel to Arduino and from Arduino to a computer or user interface. In communication from Dynamixel to Arduino, a 74S241N component with a half duplex system is used. While data communication from Arduino to a computer can be done in two ways, namely by using a serial data cable directly or can use a Bluetooth module. Bluetooth module used is HC-05 component.

6. Experimental Result

In the experiment, the walking quality of the robot was assessed by whether the robot fell or not while walking, straightness of walking direction, deviation of pitch angle, roll angle deviation, and whether or not the robot's footprint slipped when walking on the floor. Experiments were conducted to see the success of the control system to recognize the slope of the floor in the initial conditions and the transition conditions of the robot, so that it can provide the right and fast compensation of hip and ankle to the robot before starting the walking process. Another desired result is the determination of the right compensation ratio on the ankle and hip robot to ensure that the robot does not fall during walking. The delay strategy applied to the control system will be seen as the impact on improving the quality of walking robots.

The walking floor for Ardubio-Walker is designed like an seesaw which has a fulcrum in the center and has an angle range of -9 to 9 degrees as shown in figure 14.

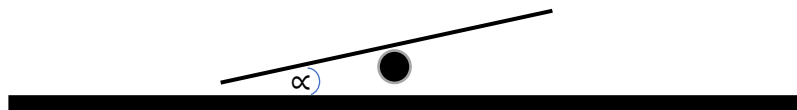


Figure 14. Side View of Walling Floor

The seesaw can be moved up and down manually to see the Ardubio-Walker response to tilt changes. In this experiment, robots are only planned to run up and down the floor, so the main concern for the control system is the change in pitch angle, and Ardubio-Walker runs with a torso that moves forward and backward during the process.

A. Dynamixel and IMU data measurement

The main measurement data as input systems are those obtained from the IMU measurements and Dynamixel measurements. The format of the measurement results is the counter value, joint d_7 - d_{18} angle (measurement), Pitch angle, and Roll angle. Examples of readings from robots in counter 1 are as follows:

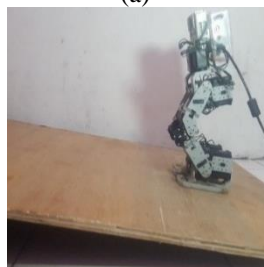
cnt	D07	D08	D09	D10	D11	D12	D13	D14	D15	D16	D17	D18	Pitch	Roll
1	354	665	554	527	281	741	30	985	800	218	571	531	82	89

By using this input data format, the current condition of the robot can be seen and can be processed by the system easily and correctly. To analyze the stability of the robot in the CoM term, a calculation is needed to find the position of CoM using DH (Denavit Hartenberg) to obtain the position of CoM at each sampling, after which the average position of CoM can be searched in one cycle. In addition to the position of the CoM, information on the angle of deviation from the robot torso is required, namely the pitch value and roll value. IMU produces pitch and roll data that requires a filtering process with Kalman Filter programming on Arduino modules, the resulting cleanliness of the signal and lack of noise is apparent.

B. Initial Inclination Detection



(a)



(b)

Figure 15. (a) Pose 0 on 8° and (b) Pose 2 on 8°

Initial slope detection is carried out in 2 conditions, namely from the sitting condition (pose 0) of the robot and from the condition of not sitting of the robot. In the experiments that have been carried out, the slope detection under initial conditions is carried out on slopes 0 to 12

degree with 10 attempts each, with a success rate of slope detection of 100%, so Ardubio-Walker is able to start the pose 2 (ready to walk pose) with a customized pose with ankle and hip compensation that matches the slope of the floor as can be seen in Figure 15.

C. Transitional Inclination Detection

Control with the transitional inclination detection approach is performed when the robot is in the double support phase (DSP). This control aims to recognize patterns from the Dynamixel position input and input the pitch value of the IMU so that it can detect the current state of the robot and read the FSA, FSH and delay correction values that are working on the robot, so as to provide further correction that matches the new floor slope .

Slope detection in transition conditions is tilt detection when tilt changes occur on the floor where the robot is walking. The results achieved by the robot can read the floor inclination with an error tolerance of ± 1 degree from the actual floor slope. However, the robot is still able to walk without falling and only experiences imperfections when stepping on its feet and causing a deviant walking direction.

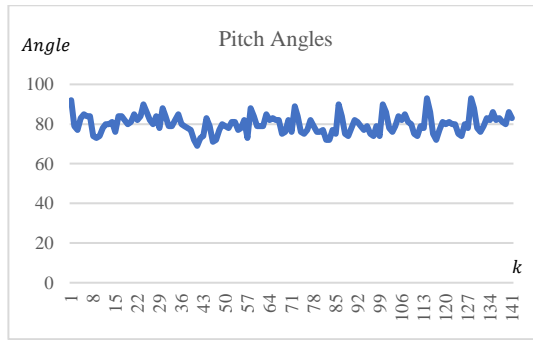
Figures 17 and 18 are the result of the Ardubio-Walker walk from a sitting pose, a standing that has adapted to the slope of the floor, and made 20 full steps until finally taking the closing step. During walking, the floor surface is moved manually continuously from a slope of 8° to -2° . Figure 19 shows that Ardubio-Walker can adapt the hip and ankle according to the given slope changes.

Figure 20 is the result of a robot from a silent pose, then a standing pose that has adjusted the slope of the floor, then walks 20 sets of full paths until it stops. During walking, the surface of the floor is changed simultaneously from the surface up 8 degrees to the surface down -2 degrees. Figure 20 shows that the robot is able to adapt to the slope of the floor by carrying out the proposed strategy.

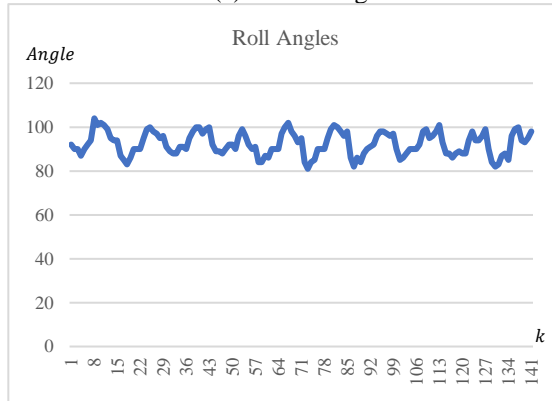


Figure 16. Walk in slope transition

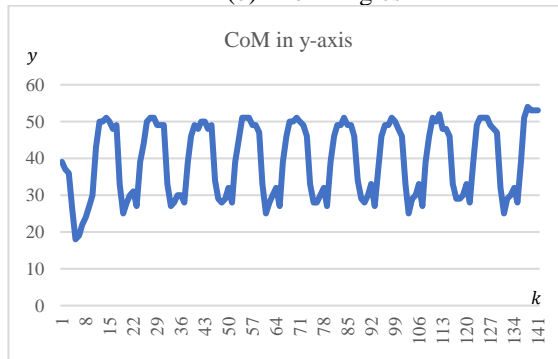
Figure 20 shows the result of the experiment for the robot to walk on 12.5 degree. The result show that robot vibrates in the beginning of walking process, but start to be stable with pitch angle mean in 85 degree and roll angle mean in 93 degree when the robot starts the second step. When the inclination is increased to 13 degree, the RBW can still walk without fall even there is a very vibrate pitch in the initial step, but the sole was slipped, that make RBW like doing walk in place motion.



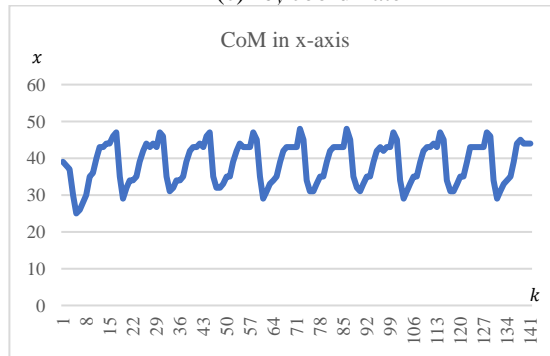
(a) Pitch Angles



(b) Roll Angles



(c) c_y coordinate



(d) c_x coordinate

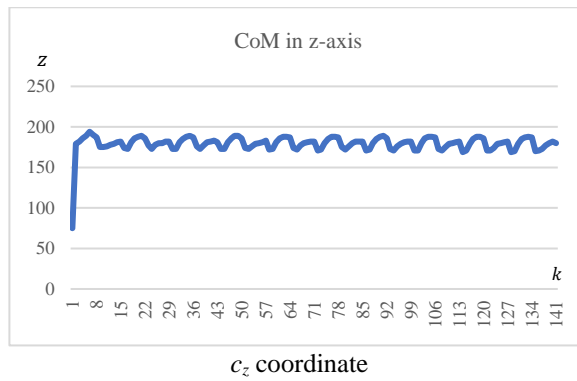


Figure 17. (a)-(e) Initial to transition walk graphs with 8° to -2° inclination

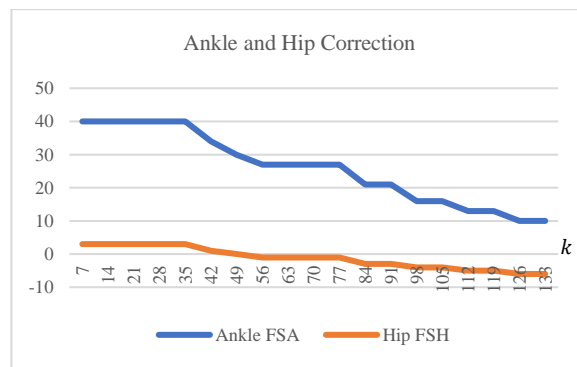


Figure 18. Hip and Ankle Correction value while walking 20 steps from inclination 8° to -2°

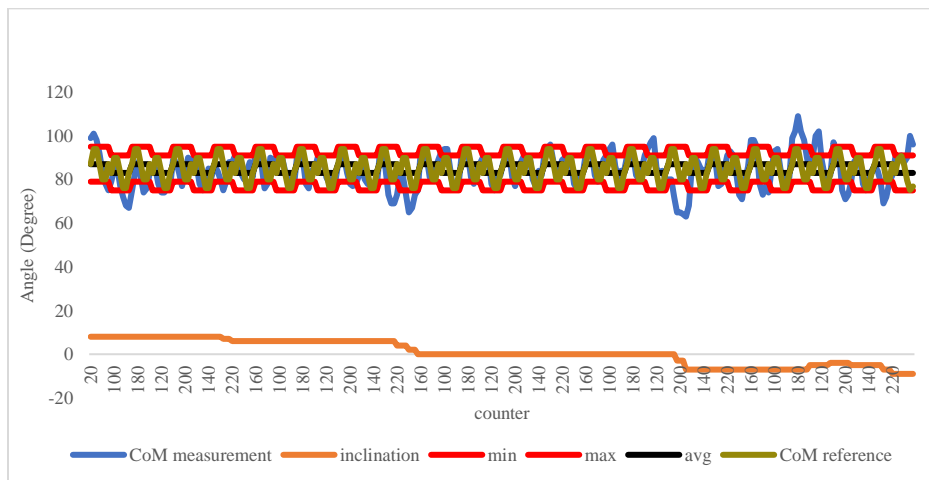


Figure 19. CoM reference, CoM measurement, CoM mean, stability margin, and Inclination measured

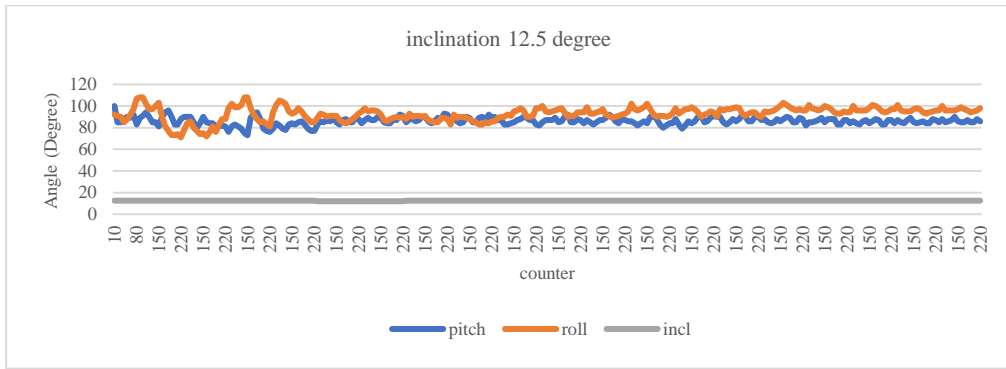


Figure 20. Pitch and Roll angle and inclination measurement for experiment with inclination 12.5 degree

D. Ankle and Hip Strategies

Ankle and hip compensation are given to maintain a balanced state of the robot is carried out by conducting an ankle and hip compensation distribution with a certain ratio. The quality of walking is judged by whether the robot falls or not, slips or not, straight walk or not, and the amount of pitch and roll angle deviation when walking. Each parameter has a value of 20% out of a total of 100% the value of running quality. Experiments in distributing compensation to the ankle and hip were carried out through experiments for 20 step set at a slopes with the following results.

Table 4. Experiments at an angle of 6 degree

Ankle	Hip	Fall/Not	Straight Walk/Not	Slip/Not	Pitch Range	Roll Range	Walk Quality
100%	0%	Not fall	Straight walk	Not Slip	21	20	100%
75%	25%	Not fall	Straight walk	Not Slip	24	23	100%
70%	30%	Not Fall	Straigh walk	Not Slip	18	22	100%
50%	50%	Not Fall	Straight walk	Not Slip	18	23	100%
25%	75%	Fall	---	---	---	---	0%
0%	100%	Fall	---	---	---	---	0%

Table 5. Experiments at an angle of 8 degree

Ankle	Hip	Fall/Not	Straight Walk/Not	Slip/Not	Pitch Range	Roll Range	Walk Quality
100%	0%	Not fall	Straight walk	Not Slip	25	21	100%
75%	25%	Not fall	Straight walk	Not Slip	26	21	100%
70%	30%	Not Fall	Straigh walk	Not Slip	20	22	100%
50%	50%	Not Fall	Not Straight	Not Slip	22	23	80%
25%	75%	Fall	---	---	---	---	0%
0%	100%	Fall	---	---	---	---	0%

Table 6. Experiments at an angle of 10 degree

Ankle	Hip	Fall/Not	Straight Walk/Not	Slip/Not	Pitch Range	Roll Range	Walk Quality
100%	0%	Not fall	Not Straight	Slip	24	23	60%
75%	25%	Not fall	Straight walk	Not Slip	23	21	100%
70%	30%	Not Fall	Straigh walk	Not Slip	18	22	100%
50%	50%	Not Fall	Not Straight	Not Slip	18	22	80%
25%	75%	Fall	---	---	---	---	0%
0%	100%	Fall	---	---	---	---	0%

Table 7. Experiments at an angle of 12 degree

Ankle	Hip	Fall/Not	Straight Walk/Not	Slip/Not	Pitch Range	Roll Range	Walk Quality
100%	0%	Not fall	Not Straight	Slip	38	27	20%
75%	25%	Not fall	Straight	Not Slip	34	33	60%
70%	30%	Not Fall	Straight	Not Slip	22	23	100%
50%	50%	Not Fall	Not Straight	Not Slip	23	18	80%
25%	75%	Fall	---	---	---	---	0%
0%	100%	Fall	---	---	---	---	0%

A summary of the results of a combination of hip and ankle strategies with certain ratios can be seen in the bar diagram in the Figure 21 below. From the bar chart it can be seen that the combination of strategies shows maximum results at the ankle and hip ratio of 70%: 30% followed by a ratio of 75%: 25%.

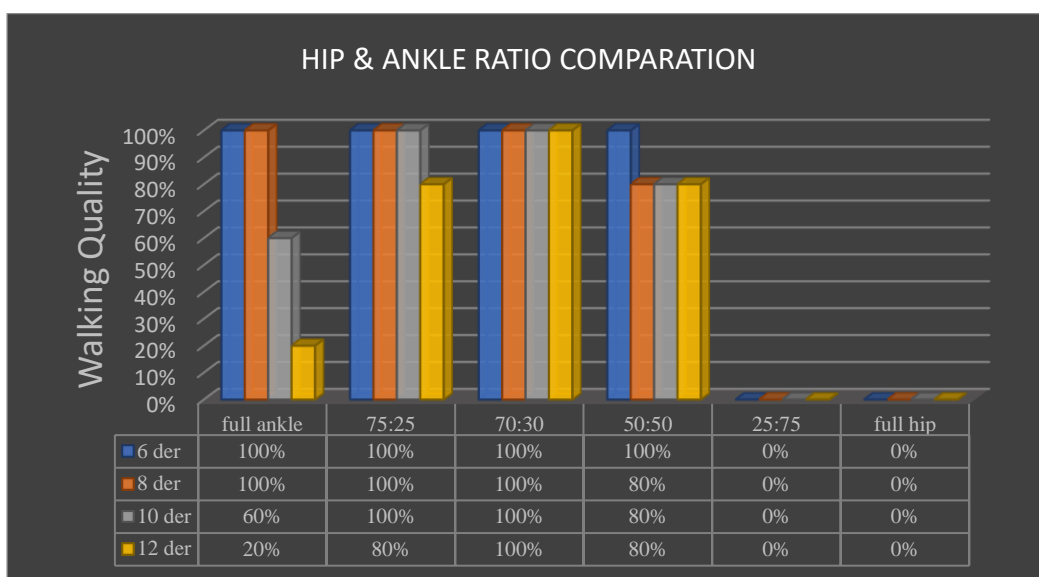


Figure 21. Summary of the experiment with hip and ankle ratio

E. Delay Strategy

In the delay strategy experiment, experiments were conducted with scenarios: without giving delay, giving delay with uniform values, and giving values with varying values. The experiment was carried out at a slope of 8 degree with the ratio of ankle and hip strategy that is used is 75%:25%. The Result of the experiment can be seen in Table 8.

Table 8. Delay Time Experiments at 8 Degree with Ratio 75% : 25%

Delay Time	Fall/Not	Straight /Not	Slip/Not	Pitch Range	Roll Range
0	Not Fall	Not Straight	Not Slip	28	24
5	Not Fall	Not Straight	Not Slip	25	24
10	Not Fall	Straight	Not Slip	22	20
20	Not Fall	Straight	Not Slip	22	19
25	Not Fall	Straight	Not Slip	23	23
30	Not Fall	Straight	Not Slip	28	16

7. Conclusion

From the results of the experiments conducted, it can be shown that the slope detection strategy is capable of detecting slope well, so as to be able to prepare balanced ready-to-walk

poses on the Bioloid Walk. The compensation of the ankle and hip successfully maintains the balance of the robot when walking with a ratio of 50% to 100% in the ankle, to the percentage of the ankle below 50% on the slope, unable to maintain the balance of the robot on slope and fall in the third step. Consistent better quality results are shown for a ratio of 70% for the ankle and 30% for the hip which results in a walking balance with a greater slope difference. Delay strategy can reduce the pitch and roll margin when it is given at the slope. The delay that gives the better result for roll and pitch range is 1 or 2 times of inclination. The more delay makes the robot like doing a static walking motion. With the proposed transition strategy, the robot is able to adapt by adjusting the hip and ankle to walk up and down, with the same ratio, adjusting to the slope of the floor.

8. Reference

- [1] Riyanto, W. Adiprawita, H. Hindersah and C. Machbub, "Center of Mass based Walking Pattern Generator with Gravity Compensation for Walking Control on Bioloid Humanoid Robot," *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, Singapore, 2018, pp. 54-59. doi: 10.1109/ICARCV.2018.8580633
- [2] Prahlad Vadakkepat, Dip Goswami and Chia Meng Hwee, "Disturbance Rejection by Online ZMP Compensation", *Robotica* 26, (2008) 9{17.
- [3] Horak and L. Nashner, "Central programming of postural movement: adaptation to altered support-surface configurations", *Journal of Neurophysiology* 55(6),(1986) 1369{1381.
- [4] H. Hemami and P. Camana, "Nonlinear feedback in simple locomotion systems", *IEEE Transactions on Automatic Control* 21(6),(1976) 855{860.
- [5] Q. Huang, K. Kaneko, K. Yokoi et.al., "Balance Control of a Biped Robot Combining Online Pattern with Real-time Modification", *Proceedings of the 2000 IEEE International Conference on Robotics and Automation* (2000) pp. 3346{3352.
- [6] Kim, J.-W., Tran, T. T., Van Dang, C., & Kang, B. (2016). Motion and Walking Stabilization of Humanoids Using Sensory Reflex Control. *International Journal of Advanced Robotic Systems*, 13(2), 77. doi:10.5772/63116 Singapore, 2018, pp. 36-41. doi: 10.1109/ICARCV.2018.8581061
- [7] Dragomir N. Nenchev and Akinori Nishio," Ankle and hip strategies for balance recovery of a biped" *Robotica* (2008) volume 26, pp. 643–653. © 2008 Cambridge University Pressdoi:10.1017/S0263574708004268
- [8] Guihard, M., & Gorce, P. (n.d.). Dynamic control of bipeds using ankle and hip strategies. *IEEE/RSJ International Conference on Intelligent Robots and System*. doi:10.1109/irids.2002.1041660
- [9] Aftab, Zohaib & Robert, Thomas & Wieber, Pierre-Brice. (2012). Ankle, Hip and Stepping Strategies for Humanoid Balance Recovery with a Single Model Predictive Control Scheme. *IEEE-RAS International Conference on Humanoid Robots*. 10.1109/HUMANOIDS.2012.6651514.
- [10] Martin, D. I. H. Putri, Riyanto and C. Machbub, "Gait Controllers on Humanoid Robot Using Kalman Filter and PD Controller," *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*,
- [11] http://support.robotis.com/en/product/actuator/Dynamixel/ax_series/dx1_ax_actuator.htm, accessed on March 12, 2019
- [12] www.arduino.cc, accessed on March 12, 2019
- [13] Kalman, R. E. 1960. "A New Approach to Linear Filtering and Prediction Problems," *Transaction of the ASME-Journal of Basic Engineering*, pp. 35-45 (March 1960).
- [14] Miranda, J. L. C. (2009): Application of Kalman filtering and PID control for. *Master Thesis California State University*, 103
- [15] Philippe Miranda. 2017. "Algorithms of Inertial Measurement Units based on Kalman Filter," Internship report at Texys.FR Varennes-Vauzelles – France.



Riyanto was born in Sukoharjo, Center of Java, Indonesia in 1977. He is a Ph.D student in School of Electrical Engineering and Informatics, Institut Teknologi Bandung. He obtained his Bachelor Degree in Electrical Engineering, Gajah Mada University, Indonesia in 2004, and his Master Degree in Electrical Engineering, Universitas Indonesia in 2012. Currently, he serves as governmental researcher in Agency for The Assesment and Application of Technology (BPPT), Indonesia



Carmadi Machbub got Bachelor degree in Electrical Engineering from the Institut Teknologi Bandung (ITB) in 1980, Master degree (DEA) in Control Engineering and Industrial Informatics in 1988, and Doctorate degree in Engineering Sciences majoring in Control Engineering and Industrial Informatics from Ecole Centrale de Nantes in 1991. He is now Professor and Head of Control and Computer Systems Research Division, School of Electrical Engineering and Informatics, ITB. His current research interests are in machine perception and control.



Hilwadi Hindersah graduated his Doctoral program from Institut Teknologi Bandung. His employment as a lecturer and researcher in School of Electrical Engineering and Informatics ITB. His fields of interest are vision-based robot control, interaction design, and adaptive control



Widyardana Adiprawita. Assistant professor at STEI ITB. Received his Electrical Engineering degree in Electrical Engineering ITB with honor in 1997. Finished master degree at Informatics Engineering ITB in 2000, and Finished doctorate degree at 2011 with honor in Electrical Engineering and Informatics ITB. His research interests are embedded system, robotics, computer vision, and artificial intelligent. He has written more than 50 papers published in international publication.