Development of a Biped Humanoid Robot - BabyBot

Yonghwan Oh, Youngjin Choi, Bum-Jae You, and Sang Rok Oh

Intelligent Robotics Research Center,
Korea Institute of Science and Technology (KIST)
39-1, Hawolgok-Dong, Sungbuk-Gu, 136-791, Korea
Tel : +82-2-958-5756, Fax : +82-2-958-5749
{o@, c@, b@, s@}@kist.re.kr

Abstract: The humanoid robot has a large number of actuators to mimic the human behaviors, e.g., BabyBot developed as one of humanoid platforms consists of 24 electric motors. In this case, the decentralized controllers such as PID ones are used to control the position of motors, since it is hard for complex centralized controllers to use. Moreover, although the controller has the decentralized type, it takes much time to tune the motion control performance of humanoid robot since the actuators show the different characteristics according to their capacity. In order to achieve the target performance quickly, BabyBot uses the decentralized auto-tuning PID controller. In this paper, the auto-tuning criterion for BabyBot is suggested to satisfy the target performance. Finally, experimental results show that the target performance can be quickly accomplished by it while executing tasks.

Keywords: Humanoid Robot, Auto-tuning Control, Target Performance, Criterion of Auto-tuning

1. Introduction

The final goal of robotics is the humanoid robot which can mimic the human behaviors, intelligence and interaction. Although the intelligence among them will be the core technology of the humanoid robot in future, it is currently important for the platform technology concerning robotic motion and interaction in developing the humanoid robot. A variety of platforms for humanoid robot should be developed to realize the intelligence, but most researchers still have difficulties in developing the hardware platform before realizing intelligence. Especially, the humanoid robot uses a large number of electric motors to mimic the human behaviors. Actually, it takes much time to tune the gain values of controller since so many and various motors are used in humanoid robot. Also, the individually tuned control gains does not guarantee that they can satisfy the target performance in total motors coordination of humanoid robot. In this respect, auto-tuning controller is required to satisfy the target performance automatically and quickly.

As a matter of fact, there are many references [1], [2] about adaptive controllers which can show the good performance for robotic systems. Since they require the complex dynamics to use (especially, the humanoid robot dynamics are so complex), however, most researchers prefer the decentralized PID controllers to centralized dynamic ones because the decentralized PID ones are simple to use. Recently, the decentralized auto-tuning PID controller was suggested in [3], [4], by using the direct adaptive control method for an inverse optimal PID controller suggested in [5]. Also, if several conditions for control gains are satisfied, then the decentralized auto-tuning PID controller has the extended disturbance input-to-state stability for robotic systems. The BabyBot uses the decentralized auto-tuning PID controller in order to achieve the target performance quickly. Also, how to make the target performance numerically will be discussed in this paper.

This paper is organized as follows: section 2 introduces the humanoid robot BabyBot in view of the design concept, kinematics and control strategy, section 3 suggests how to make target performance of decentralized auto-tuning PID controller, section 4 shows the superiority
2. Humanoid Robot System: BabyBot

The humanoid robot has been developed to mimic the human behaviors, intelligence and interaction. Actually, it is possible for humanoid robot to serve more functions than the mobile robot. Now, we are developing the BabyBot as a humanoid robot platform, its design concept and control structure are explained in detail in following sections.

2.1. Design Concept

In order to design a humanoid robot system, we should first determine its physical dimension. Those kinematic data were obtained from one-year old Korean standard baby, e.g., the height of BabyBot was chosen to be 75 cm because the Korean standard height was 74.6 cm and 76.4 cm for one-year old boy and girl, respectively. Also, we tried to set other kinematic data (shoulder height, leg length, arm length, etc.) fit for Korean standard ones, in other words, the kinematic Denavit-Hartenberg parameters were determined according to the Korean standard ones.

The photograph of BabyBot is shown in Fig. 1. The BabyBot has 24 degrees-of-freedom (DOF) in total, concretely speaking, the neck motion system has 2 DOF, each arm has 5 DOF and each leg has 6 DOF as shown in Fig. 2. Also, the Denavit-Hartenberg parameters of dual arms and legs are listed in Table 1 and 2. The BabyBot has the functions such as voice recognition and speaking module which can recognize and respond to 35 voice commands, e.g., go forward/backward, turn left/right, say hello/introduction, dance etc. The BabyBot draws the conclusion.

Table 1. Kinematic link parameters of dual manipulator system

<table>
<thead>
<tr>
<th>Link</th>
<th>( \theta_i )</th>
<th>( d_i )</th>
<th>( \alpha_i )</th>
<th>( \alpha_i )</th>
<th>( \theta_{i,0} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( \phi_{11} )</td>
<td>(- \ell_S )</td>
<td>0</td>
<td>( \pi/2 )</td>
<td>(- \pi/2 )</td>
</tr>
<tr>
<td>2</td>
<td>( \phi_{12} )</td>
<td>0</td>
<td>0</td>
<td>( -\pi/2 )</td>
<td>( \pi/2 )</td>
</tr>
<tr>
<td>3</td>
<td>( \phi_{13} )</td>
<td>(- \ell_U )</td>
<td>( \ell_E )</td>
<td>( \pi/2 )</td>
<td>( \pi/2 )</td>
</tr>
<tr>
<td>4</td>
<td>( \phi_{14} )</td>
<td>0</td>
<td>( \ell_L )</td>
<td>( \pi/2 )</td>
<td>(- \pi/2 )</td>
</tr>
<tr>
<td>5</td>
<td>( \phi_{15} )</td>
<td>0</td>
<td>( \ell_H )</td>
<td>( \pi )</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>( \phi_{r1} )</td>
<td>\ell_S</td>
<td>0</td>
<td>(- \pi/2 )</td>
<td>(- \pi/2 )</td>
</tr>
<tr>
<td>2</td>
<td>( \phi_{r2} )</td>
<td>0</td>
<td>0</td>
<td>( \pi/2 )</td>
<td>( \pi/2 )</td>
</tr>
<tr>
<td>3</td>
<td>( \phi_{r3} )</td>
<td>\ell_U</td>
<td>( \ell_E )</td>
<td>(- \pi/2 )</td>
<td>( \pi/2 )</td>
</tr>
<tr>
<td>4</td>
<td>( \phi_{r4} )</td>
<td>0</td>
<td>( \ell_L )</td>
<td>(- \pi/2 )</td>
<td>(- \pi/2 )</td>
</tr>
<tr>
<td>5</td>
<td>( \phi_{r5} )</td>
<td>0</td>
<td>( \ell_H )</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

\( \ell_S = 0.142, \ell_U = 0.105, \ell_L = 0.1, \ell_H = 0.08 \) and \( \ell_E = 0.0025 \)
Table 2. Kinematic link parameters of leg system

<table>
<thead>
<tr>
<th>Link</th>
<th>Left / Right leg</th>
<th>$\theta_i$</th>
<th>$d_i$</th>
<th>$a_i$</th>
<th>$\alpha_i$</th>
<th>$\theta_{i,0}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$q_1/q_1$</td>
<td>0</td>
<td>0</td>
<td>$\pi/2$</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$q_2/q_2$</td>
<td>0</td>
<td>0</td>
<td>$-\pi/2$</td>
<td>$-\pi/2$</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$q_3/q_3$</td>
<td>$\ell_1$</td>
<td>$\pi/2$</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$q_4/q_4$</td>
<td>$\ell_2$</td>
<td>$-\pi/2$</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>$q_5/q_5$</td>
<td>0</td>
<td>0</td>
<td>$\pi/2$</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>$q_6/q_6$</td>
<td>$\ell_3$</td>
<td>$-\pi/2$</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$\ell_1 = 0.13$, $\ell_2 = 0.13$ and $\ell_3 = 0.04$

Table 3. Actuator system for BabyBot

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Leg J1</td>
<td>42.5</td>
<td>492:1</td>
<td>2657W-024CR</td>
<td>0.0348</td>
</tr>
<tr>
<td>Leg J2</td>
<td>42.5</td>
<td>200:1</td>
<td>2657W-024CR</td>
<td>0.0348</td>
</tr>
<tr>
<td>Leg J3</td>
<td>42.5</td>
<td>250:1</td>
<td>2657W-024CR</td>
<td>0.0348</td>
</tr>
<tr>
<td>Leg J4</td>
<td>42.5</td>
<td>200:1</td>
<td>2657W-024CR</td>
<td>0.0348</td>
</tr>
<tr>
<td>Leg J5</td>
<td>42.5</td>
<td>252.5:1</td>
<td>2657W-024CR</td>
<td>0.0348</td>
</tr>
<tr>
<td>Leg J6</td>
<td>42.5</td>
<td>202:1</td>
<td>2657W-024CR</td>
<td>0.0348</td>
</tr>
<tr>
<td>Arm J1</td>
<td>23.2</td>
<td>101.5:1</td>
<td>2642W-024CR</td>
<td>0.0346</td>
</tr>
<tr>
<td>Arm J2</td>
<td>23.2</td>
<td>99:1</td>
<td>2642W-024CR</td>
<td>0.0346</td>
</tr>
<tr>
<td>Arm J3</td>
<td>6</td>
<td>157:1</td>
<td>RE-max 21</td>
<td>0.0250</td>
</tr>
<tr>
<td>Arm J4</td>
<td>2.64</td>
<td>159:1</td>
<td>1724T-024SR</td>
<td>0.0264</td>
</tr>
<tr>
<td>Arm J5</td>
<td>2.09</td>
<td>134:1</td>
<td>1717T-024SR</td>
<td>0.0168</td>
</tr>
<tr>
<td>Head J1</td>
<td>2.64</td>
<td>246:1</td>
<td>1724T-024SR</td>
<td>0.0264</td>
</tr>
<tr>
<td>Head J2</td>
<td>2.09</td>
<td>159:1</td>
<td>1717T-024SR</td>
<td>0.0168</td>
</tr>
</tbody>
</table>

All encoders have 512 [pulse/rev]

Bot will be equipped with the camera and image processing module to realize the visual intelligence. Also, the BabyBot has the gyroscope to control the posture by oneself.

2.2. Control Structure

The DC servo motors are used as actuators to drive humanoid system. The motors (manufactured by Maxon or Minimotor Co.) and planetary gear-head are used to drive upper body system composed of dual arms and neck motion system. Also, the motors and harmonic drive systems are selected for leg systems. Table 3 shows detail specification of selected actuator system.

To control and coordinate the motion of humanoid robot, the local controllers were designed as small as possible by using DSP’s (TMS320VC33 manufactured by TI Co.) and motor amplifier chips as shown in Fig. 3. Also, the information exchange between DSP’s is achieved by the CAN communication network with the layout as shown in Fig. 4. Main DSP module gathers the information from voice recognition module, gyroscope and local DSP’s covering the motion control of arms, legs, neck motion system. And then, it orders the desired cartesian positions and velocities to local DSP’s by using CAN communication. All these modules are embedded in the abdomen and backpack of BabyBot in Fig. 1.

As shown in Table 3, since a variety of motors and gear ratios are used in humanoid robot, it takes much time to tune the control gain values, though the decentralized PID controller are simply used for the humanoid robot. As an alternative of this problem, the auto-tuning method for control gains and auto-tuning criterion are suggested in the following section.

3. Automatic Tuning for Motion Control

Performance

The inverse optimality of PID controller for mechanical systems was proved for the first time in [5]. Since it requires some conditions for gain values, it is called as
“inverse optimal PID controller” which has the following form:

\[
\tau_i = \left( k_i + \frac{1}{\gamma^2} \right) s_i 
\]
\[
s_i = \dot{e}_i + k_P e_i + k_I \int e_i dt \quad (1)
\]
\[
\text{for } i = 1, 2, \cdots , n,
\]
and its design conditions are as follows:

(C1) \( k_i > 0, k_P > 0, k_I > 0, \gamma > 0 \),

(C2) \( k_P^2 > 2k_I \),

where \( \tau_i \) means the torque control input, \( s_i \) is the composite error, \( \dot{e}_i \) expresses the velocity error, position (or configuration) error and integration error for \( i \)-th actuator, respectively and \( n \) means the number of actuators, e.g., \( n = 24 \) for BabyBot. According to the layout of actuators in BabyBot, each actuator requires the different control gains to guarantee the constant control performance. Actually, it seems to be impossible to tune the individual control gains with guaranteeing the same control performance because the BabyBot has 72 gain tuning variables (24 actuators \( \times \) 3 PID gains) in total. As an alternative, the automatic performance tuning method of an inverse optimal PID controller (1) was suggested in [3] as follows:

\[
\frac{dk_i}{dt} = \Gamma_i s_i^2 \quad \text{for } i = 1, 2, \cdots , n. \quad (3)
\]

where \( k_i(t) \) is auto-tuning gain parameter of \( i \)-th controller and \( \Gamma_i \) is the update gain parameter which determines the updating rate of auto-tuning gain parameter of \( i \)-th controller. As a matter of fact, the variation of gain \( k_i(t) \) brings that of all PID gains in aspect of conventional PID controllers as follows:

- **P control**: \( (k_i(t) + \gamma^{-2}) \cdot k_P \cdot e_i \),
- **I control**: \( (k_i(t) + \gamma^{-2}) \cdot \int e_i dt \),
- **D control**: \( (k_i(t) + \gamma^{-2}) \cdot k_I \cdot \dot{e}_i \).

In following section, the criterion for auto-tuning will be suggested to guarantee the constant control performance which is defined as ‘target performance’.

### 3.1. Target Performance

Actually, there exist the gain values guaranteeing target performance (\( \Omega_i \)) for each actuator, however, we can not know the gain value \( k_i \) till the experimental result satisfies the target performance as follows:

\[
\sup_{0 \leq t \leq T_f} |s_i(t)| \leq \Omega_i,
\]

because the required control inputs are different according to the layout of actuators in BabyBot. To satisfy the target performance, the auto-tuning law should be applied to the smaller composite error than the target performance. Also, since the auto-tuning law (3) was composed of the decentralized type, we suggest the decentralized criterion for auto-tuning as follows:

\[
|s_i| > \frac{\Omega_i}{\sqrt{2}} \quad (4)
\]

In Fig. 5, the relationship between the tuning region and target performance was described for a simple example of \( n = 2 \). As soon as the composite error arrives at the tuning region of (4), the auto-tuning law should be implemented to assist the achievement of target performance. On the contrary, if the composite error stays in non-tuning region of Fig. 5, namely \( |s_i| \leq \frac{\Omega_i}{\sqrt{2}} \), then the auto-tuning process is stopped. For this case, we expect that the gain \( k_i(t) \) updated by an auto-tuning law (3) will be larger than the gain value which brings the target performance. As a matter of fact, the auto-tuning law has the property of a nonlinear damping. Strictly speaking, the first term of auto-tuning PID controller (1) means the nonlinear damping which helps to stabilize the control system against disturbances and the second term is a linear controller.

### 4. Experimental Results

In order to find the control gain values guaranteeing the target performance of BabyBot as shown in Fig. 1, we apply the auto-tuning method (3) to inverse optimal PID controller (1). The desired motions consist of bipedal walking, arm swing and neck swing which mimic the human behavior.

First, we determine the static gains of auto-tuning inverse optimal PID controller as \( k_P = 20, k_I = 100 \) and \( \gamma = 0.5 \) satisfying the design conditions in section 3. Then the controller has the following form: for
i = 1, 2, · · · , n,
\[\tau_i = (k_i(t) + 4)s_i\]
\[s_i = \dot{e}_i + 20e_i + 100 \int e_i dt\]
\[\frac{dk_i}{dt} = \Gamma_i s_i^2, \quad \text{if } |s_i| > \frac{\Omega_i}{\sqrt{2}}\]
\[\frac{dk_i}{dt} = 0, \quad \text{else}\]

where initial values of auto-tuning parameters \(k_i(0) = 1.0\), update gain values \(\Gamma_i = 500\) for all \(i\) and \(n = 24\). Second, since the composite error is approximately proportional to the configuration error with proportional constant \(k_p\), the target performance can be approximately determined as follows:
\[\Omega_i \approx k_p \times |e_i|t.\]  \hspace{1cm} (5)

where \(|e_i|t\) means the position (configuration) error. For instance, if we are to obtain the performance of \(|e_i|t < 0.025 [\text{rad}]\) for each driving actuator, then the target performance should be determined as \(\Omega_i = 0.5\) by (5).

Fig. 6, 7 and 8 show experimental results such as the variation of auto-tuning parameter, composite error and position (configuration) error. In figures, we can see that the errors are large at initial time and errors are reduced till target performance can be achieved by an automatic gain tuning. As a matter of fact, the automatic gain tunings are executed at the exterior of two dotted lines \(\pm 0.35 (= \Omega_i/\sqrt{2} = 0.5/\sqrt{2})\) of figures (c) and (d) of all experimental results. After the auto-tuning processes for a left leg are finished as shown in Fig. 6.(a), the auto-tuned gains are obtained as follows:
\[k_1 = 1.0 \rightarrow 21.19\]
\[k_2 = 1.0 \rightarrow 9.167\]
\[k_3 = 1.0 \rightarrow 13.08\]
\[k_4 = 1.0 \rightarrow 16.70\]
\[k_5 = 1.0 \rightarrow 13.20\]
\[k_6 = 1.0 \rightarrow 9.746.\]

Also, After the auto-tuning processes for a right leg are finished as shown in Fig. 6.(b), the auto-tuned gains are obtained as follows:
\[k_7 = 1.0 \rightarrow 16.01\]
\[k_8 = 1.0 \rightarrow 8.944\]
\[k_9 = 1.0 \rightarrow 13.19\]
\[k_{10} = 1.0 \rightarrow 14.77\]
\[k_{11} = 1.0 \rightarrow 12.64\]
\[k_{12} = 1.0 \rightarrow 9.058.\]

As we can see in Fig. 6.(a) and (b), the automatic gain tuning processes are mainly accomplished in time region of \(2.5 \sim 3.5[s]\) for a left leg and \(0 \sim 1.0[s]\) for a right leg because of the walking pattern. The walking process consists of ‘right step’ and then ‘left step’, repeatedly. Here, we can know that each stepping leg basically requires the larger gains than the supporting leg in constant walking pattern. Also, since the composite error and configuration error stay in \(|s_i(t)| < \Omega_i = 0.5\) and \(|e_i(t)| < |e_i|t = 0.025[\text{rad}]\) as shown in Fig. 6. (c)∼(f), respectively, we can know that the target performance is achieved for left and right legs.

After the auto-tuning processes for a left arm are finished as shown in Fig. 7.(a), the auto-tuned gains are obtained as follows:
\[k_{13} = 1.0 \rightarrow 2.496\]
\[k_{14} = 1.0 \rightarrow 1.0\]
\[k_{15} = 1.0 \rightarrow 1.0\]
\[k_{16} = 1.0 \rightarrow 1.0\]
\[k_{17} = 1.0 \rightarrow 1.0.\]

Also, After the auto-tuning processes for a right arm are finished as shown in Fig. 7.(b), the auto-tuned gains are obtained as follows:
\[k_{18} = 1.0 \rightarrow 2.094\]
\[k_{19} = 1.0 \rightarrow 1.0\]
\[k_{20} = 1.0 \rightarrow 1.0\]
\[k_{21} = 1.0 \rightarrow 1.0\]
\[k_{22} = 1.0 \rightarrow 1.0.\]

Also, After the auto-tuning processes for neck motion system are finished as shown in Fig. 8.(a), the auto-tuned gains obtained as follows:
\[k_{23} = 1.0 \rightarrow 1.0\]
\[k_{24} = 1.0 \rightarrow 1.0.\]

As a matter of fact, the actuators of left/right arms and neck require the smaller torque than those of legs because arms and neck do not support the whole body like legs. Hence, most controllers of arms and neck are not tuned automatically from initial gain values because the initial gains can satisfy the target performance except first controller \((k_{13}, k_{18})\) of each arm.

5. Concluding Remarks
The design concept of small-size humanoid robot (BabyBot) and its control structure based on CAN communication network were suggested in this paper. As a platform technology of humanoid robot, we suggested the control method to guarantee the constant control performance for all actuators in BabyBot. Also, we showed through experiments that the target performance could be satisfied by using the suggested control method.
Fig. 6. Experimental Results for Left/Right Legs

Table 1. Experimental Results for Left/Right Legs

<table>
<thead>
<tr>
<th>Left Leg</th>
<th>Right Leg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto-tuned Gain of Left Arm</td>
<td>Auto-tuned Gain of Right Arm</td>
</tr>
<tr>
<td>Auto-tuned Gain of Left Leg</td>
<td>Auto-tuned Gain of Right Leg</td>
</tr>
<tr>
<td>Composite Error of Left Leg</td>
<td>Composite Error of Right Leg</td>
</tr>
<tr>
<td>Configuration Error of Left Leg</td>
<td>Configuration Error of Right Leg</td>
</tr>
</tbody>
</table>

References


Fig. 7. Experimental Results for Left/Right Arms

Fig. 8. Experimental Results for Neck Motion