
SOFTWARE AND HARDWARE CONTROL OF A HYBRID ROBOT FOR SWITCHING BETWEEN LEG-TYPE AND WHEEL-TYPE MODES

Wagner Tanaka Botelho*

wagtanaka@gmail.com

Abeer Mahmoud*

abeer_f13@yahoo.com

Tokuji Okada†

okada@eng.niigata-u.ac.jp

Toshimi Shimizu†

toshimi@eng.niigata-u.ac.jp

*Course of Information Science and Engineering
Graduate School of Science and Technology, Niigata University
Niigata, Japan

†Department of Biocybernetics
Faculty of Engineering, Niigata University
Ikarashi 2-8050
Niigata, Japan, 950-2181

ABSTRACT

One of the objectives of the paper is to describe the hybrid robot PEOPLER-II (Perpendicularly Oriented Planetary Legged Robot) with regard to switching between leg-type and wheel-type. Our robot has an easier design and control system than other hybrid robots. The software and hardware control in the process of performing five robot tasks are considered. These are the walking, rolling, switching, turning and spinning. In the switching task, we show the control method based on minimization of total energy cost. Also, the hardware components and their interconnections are described. The graphical user interfaces utilized in the simulation and experiment are demonstrated. The walking, rolling and the switching without reverse rotation and arm motion are verified in simulation and with real robot, in addition to turning and spinning.

KEYWORDS: Mobile robots, legged robots, wheeled robots, hybrid robots.

RESUMO

Controle de Software e Hardware de um Robô Híbrido na Mudança no Modo de Locomoção Utilizando Pernas ou Rodas

O objetivo principal deste trabalho é descrever a mudança no modo de locomoção entre pernas e rodas do robô híbrido PEOPLER-II (*Perpendicularly Oriented Planetary Legged Robot*). A diferença entre PEOPLER-II e outros robôs híbridos está relacionado com a facilidade que foi projetado e também o seu sistema de controle. O robô realizará cinco tarefas: o caminhar, a locomoção por rodas, o chaveamento entre os modos de locomoção com pernas e rodas, o movimento de virar para esquerda ou direita e a rotação no sentido horário ou anti-horário. O chaveamento é realizado utilizando um método que reduza o custo total de energia necessária na troca entre pernas e rodas. Neste artigo serão apresentados os algoritmos de controle definidos para cada tarefa, as especificações do hardware definidas na arquitetura do robô e as interfaces do software utilizadas na simulação e no experimento. Os resultados experimentais obtidos validam a metodologia proposta.

PALAVRAS-CHAVE: Robôs móveis, perna, roda, robôs híbridos.

Artigo submetido em 18/11/2009 (Id.: 01079)

Revisado em 02/02/2010, 13/07/2010, 17/09/2010

Aceito sob recomendação do Editor Associado Prof. Luis Antonio Aguirre

1 INTRODUCTION

Legged robots have been considered for a long time as effective on irregular ground conditions. The problem compliance control during leg impact and during the interaction between the robot and the environment is one of the central problems in locomotion of legged robots (Palis e Rusin, 2004). Even though these robots are capable of traveling on uneven terrains and climb stairs, their construction is complex (Akinfiyev et al., 2008). Another characteristic is their ability to use different walking gaits in response to changes in terrain, traversing speed and tasks (Still et al., 2006). A new strategy to control an one-legged robot to reduce the energy by the system is explained (Schammas et al., 2001). The application of Euler-Lagrange formulation in dynamics of a quadruped robot leg mechanism is presented in (Pizziolo et al., 2004). A hierarchic control architecture for a four legged mobile robot using fuzzy controllers and a dual-axis accelerometer is proposed in (Lima et al., 2008). The model for determination of position of a legged robot, with respect to a fixed reference system in the start of movement is considered in (Filho et al., 2004). The LegGen (Heinen e Osório, 2007) automatically controls stable gaits for legged robots. ANTON (Konyev et al., 2008), Katharina (Palis e Rusin, 2004), LAURON III (Gamann e Berns, 2002), RALPHY (Amaral et al., 1993), RHex (Balasubramanian et al., 2008), SILO-4 (Garcia e de Santos, 2006), Scout-II (Poulakakis et al., 2006) and Tekken2 (Kimura et al., 2007) are examples of legged robots.

Wheeled robot mechanical constructions are simpler than legged robots. They are efficient at high speed on flat ground. Also, their mechanism are simply with inherent advantages, such as high energy efficiency, low noise, etc (Botelho et al., 2009). Lages e Hemerly (1998) discussed the control laws for all classes of wheeled robots. Junior (2008) designed the robot with two wheels and a caster to navigate safety in a known environment with obstacles. The trajectory controllers for differential drive, wheeled robots, with the control strategies based on output feedback is presented (Borges et al., 2003). Kühne et al. (2004) proposed a full-connected, decentralized control architecture for a wheeled robot using three different types of sensors. A graphical simulator is developed to provide a computational graphical tool to assists the design and analysis of control laws of wheeled mobile robots is presented in (Schroueder et al., 2005). Examples of wheeled robots are described in Seelinger et al. (2002), Kim e Tsiotras (2002), Khoh e Cho (1999), Lauria et al. (2002), Lapierre et al. (2007) and Woo et al. (2007). Each robot has the name of FIDO, Khepera, LCAR, OCTOPUS, Pekee, WMR, respectively.

Legged and wheeled robots have benefits and drawbacks. Therefore, hybrid robots can benefit from both legs and

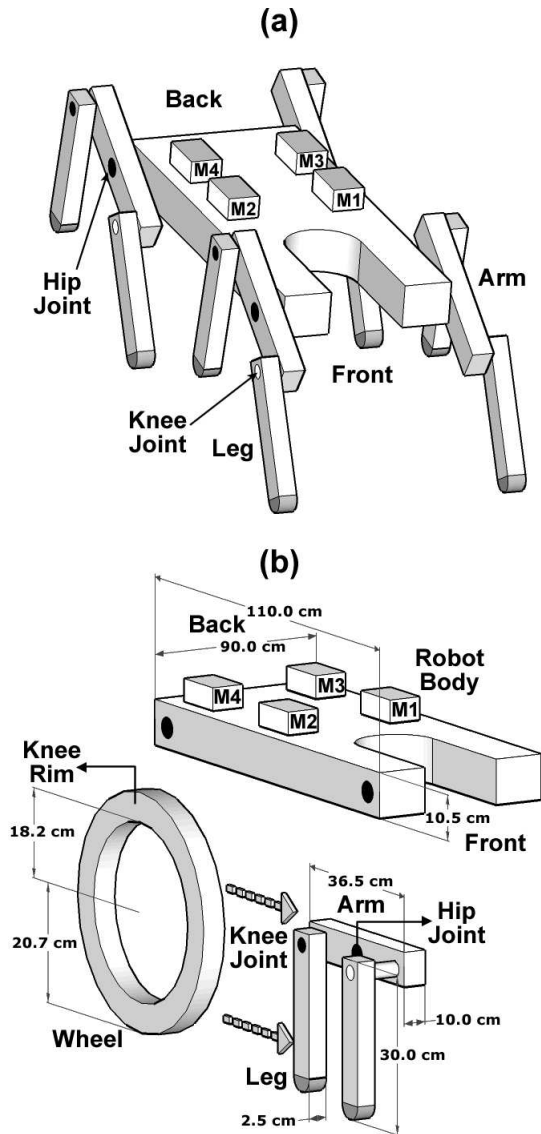


Figure 1: Legged robot PEOPLER-I and the designed notification for PEOPLER-II in (a) and (b), respectively.

wheels to improve locomotion performance (Sonehara et al., 2004). The environmental characteristics are one of the most influential aspects to be considered in our robot. While legged locomotion would more adaptable in a wide range of terrains, wheeled locomotion is faster but only on smooth surfaces. ALDURO (Germann et al., 2005), ChariotII (Dai et al., 1995), HyLoS-II (Grand et al., 2004), Roller-Walker (Endo e Hirose, 1999), Walk'n Roll (Adachi et al., 1999), Wheelleg (Lacagnina et al., 2003), WorkPartner (Aarne et al., 2001) and the wall climbing robot (Fu et al., 2007) are examples of hybrid robots.

Wheelchairs have been in use for many years now, and they

improve the mobility of the handicapped. The traditional depot-style wheelchair found in hospitals and airports is the most common wheelchair and they are characterized for having very few adjustments and not designed for daily long-term use. However, there are some wheelchairs with many adjustable features to maximize comfort and safety to the user (Richter, 2001).

Architectural barriers still exist in many cities and buildings, and it is expensive and timing consuming, if not impossible to eliminate all of them. A wheelchair becomes useless when faced with these barriers, and as a result, there has been a number of wheelchair designs that claim to be able to climb stairs (Morales et al., 2006). The wheelchair proposed in Morales et al. (2004) was designed to enforce mechanical stability while the wheelchair is on the staircase.

Most people live, work, and play in environments that are not easily accessible to those with mobile disability. Whether indoors or outdoors, there exist man-made or natural obstacles that wheelchairs cannot easily negotiate. For example, a powered wheelchair is able to negotiate curbs up to 130 mm in height with the use of a rocking arm which engages the curb and lifts the front wheels up (Browning et al., 1996).

It has been known that hybrid robots have a very complex design, even more complex than legged robots. Normally, they have a complex control system and require a large number of motors. Our robot PEOPLER-I (Okada et al., 2003) can be modified to become a hybrid wheelchair robot called PEOPLER-II (Okada et al., 2006), without additional motors.

This paper describes the design concepts of the hybrid robot PEOPLER-II. Also, we define and explain five robot tasks and their respective control algorithm. These are the walking, rolling, switching between leg-type (*L-type*) and wheel-type (*W-type*), turning and spinning. The switching task is performed using the total torque minimization method (Okada et al., 2007). The hardware components that our hybrid robot is composed of and the controller interfaces utilized in the simulation and experiment are explained and shown in this paper. We present the class diagram with seven classes for the simulation and experimental interfaces. The idea used to make the communication of hardware and software control is considered. The switching task is verified in simulation and also using the PEOPLER-II, in addition to walking, rolling, turning and spinning. In the switching task, the continuity of the hip joint rotation without reversing and the constant state of the robot arm rotation are considered. These are the main results and contributions of this work.

This paper is organized as follows. Section 2 brings a brief description of the robot and the used terminology. The total torque minimization method utilized in the switching task

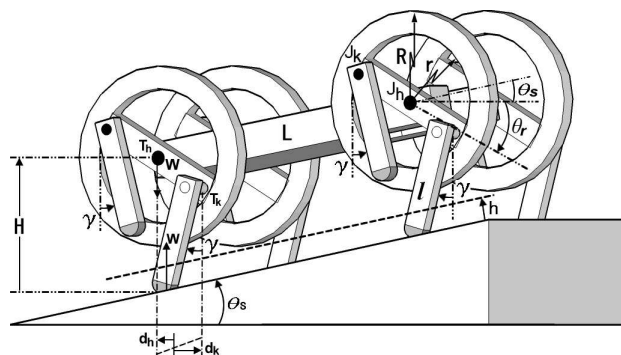


Figure 2: Nomenclatures of the new design of PEOPLER-II.

is described in Section 3. The description of the robot tasks and the control algorithms are explained in Section 4. Hardware control specifications and software control design are discussed in Section 5. Simulation and experimental results are shown in Section 6. Finally, the conclusions are presented in Section 7.

2 PEOPLER-II DESCRIPTIONS AND TERMINOLOGY

The first prototype, PEOPLER-I is a walking robot composed of 4DOFs (Degree of Freedom) utilized to walk on irregular terrain or climbing stairs as shown in Figure 1(a). Right and left sides are mechanically identical. It has four arms connected by the hip joint to each axis in the right and left robot sides. Each arm has two legs connected at the knee joint situated at the two ends of the arm. The leg posture stands for leg direction angle from the gravity direction as shown in Figure 2. The leg postures angles are same in the both directions but different in its sign because only one motor drives the two legs for saving installation cost of actuators. That is, each leg pair automatically swing opposite with mechanical symmetry in the direction of gravity.

In total, the robot has four motors (M1, M2, M3, M4) of 120 W each as shown in Figure 1. M1 and M2 are utilized to control the left and right arms. Arms in the front and back are combined mechanically. Another two motors M3 and M4 are responsible for activating two leg pairs on the left and right, respectively. Two legs pairs located at the front and back are also combined mechanically, so that they move in synchronization.

Four wheels have been attached to PEOPLER-I to transform it into the hybrid robot PEOPLER-II shown in Figure 1(b). The arrows indicate that the wheel is connected to the arm ends. Each arm acts as a central spoke supporting the wheel. Notice that the same motors are utilized to rotate the

arms and the wheels in the L-type and W-type, respectively. Therefore, additional motors are not needed. The arm is 36.5 cm long, 10.0 cm width. The 30.0 cm length with 2.5 cm width is the leg description. The outer radius of the wheel is 20.7 cm, the inner radius is 18.2 cm. The total robot weight is 107.0 kg. Finally, the robot body length, width and height are 110.0 cm, 90.0 cm and 10.5 cm, respectively. A schematic model with parameters and constants considered in this paper is shown in Figure 2. It shows the new design of PEOPLER-II on a slope standing as a L-type. Symbolic notations are as follows

- c : ratio of the hip joint motor capacity to the knee joint motor capacity;
- d_h : projected distance of hip joint from the leg end contact on the ground;
- d_k : projected distance of knee joint from the leg end contact on the ground;
- h : obstacle height;
- H : hip joint height;
- J : joint (J_h ; hip joint, J_k ; knee joint);
- l : leg length of the robot;
- L : length of the robot body (distance between front and rear hip joint axes);
- r : radius of knee joint rim;
- R : radius of a wheel expressed as a circular profile surrounding knee joints;
- T_h : torque of J_h ;
- T_k : torque of J_k ;
- W : robot weight operating at pivotal foot position;
- γ : leg posture angle (a pair of legs is symmetrical in the direction of gravity);
- θ : arm angle from the robot's front direction;
- θ_r : arm angle from the front horizontal direction ($\theta_r = \theta - \theta_s$);
- θ_s : inclination angle of the road surface.

3 METHOD FOR MINIMIZING THE TOTAL ENERGY COST

In order to make the robot stand always, we consider the kinetic related to the torques at joints J_h and J_k . Figure 2, we may derive the following equations.

$$T_h = Wd_h = W(r \cos \theta_r - l \sin \gamma), \quad (1)$$

$$T_k = Wd_k = Wl \sin \gamma, \quad (2)$$

$$H = r \sin \theta_r + l \cos \gamma. \quad (3)$$

The total energy cost, E , is evaluated by

$$E = (T_h)^2 + (cT_k)^2. \quad (4)$$

To make the value of E minimal it follows that

$$\frac{\partial E}{\partial \theta_r} = 0. \quad (5)$$

Using Eqs. 1 and 2, we find the 6th order equation of unknown parameter $\sin \theta_r$, then we have

$$\begin{aligned} \frac{\partial E}{\partial \theta_r} = & r^4(A^2 + 2A - 3) \sin^6 \theta_r - 2Hr^3(2A^2 + 3A - 5) \\ & \times \sin^5 \theta_r + r^2\{6H^2(A^2 + A - 2) - (r^2 + l^2) \\ & \times (A^2 + 2A - 3)\} \sin^4 \theta_r - 2Hr\{H^2(2A^2 + A - 3) \\ & - r^2(2A^2 + 3A - 4) - l^2(A^2 + A - 3)\} \sin^3 \theta_r \\ & + [H^2\{H^2(A^2 - 1) - r^2(6A^2 + 6A - 7) - l^2(A^2 \\ & - 2)\} - r^2\{r^2 - l^2(A^2 + 2A - 1)\} - l^4] \sin^2 \theta_r \\ & + 2Hr\{H^2(2A^2 + A - 1) + r^2 - l^2(A^2A - 1)\} \\ & \times \sin \theta_r - H^2\{H^2A^2 + r^2 - l^2A^2\} = 0, \quad (6) \end{aligned}$$

where

$$A = c^2 + 1. \quad (7)$$

Solving Eq.6 is not simple, but graphical tracing and exploration gave the angle θ_r through $\sin \theta_r$. With this, γ is calculated from Eq.3.

4 ROBOT TASK CONTROL ALGORITHM

The user, computer and PEOPLER-II have been defined as actors in our system. The user interacts with the system via command line. The computer interacts with the User via Graphical User Interface (GUI), and the PEOPLER-II is the main actor that receives tasks to be performed. The descriptions of the robot tasks and the control algorithm are described.

An activity diagram represents the state of the method execution, that is, a period of time during which one of the steps is being performed. Its purpose is to explain the algorithm involved in performing a method (Booch et al., 1996). Also, it is the object oriented equivalent of flow charts from structured development view points.

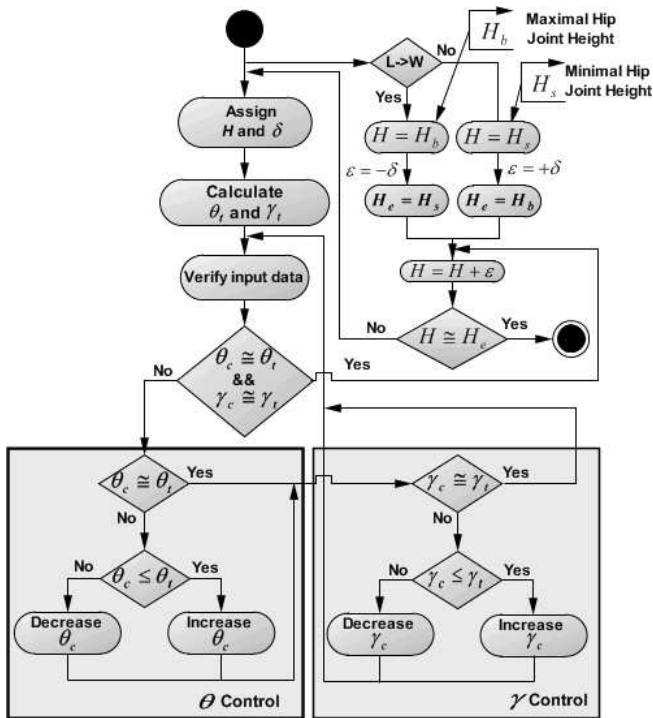


Figure 3: The activity diagram of the switching task.

4.1 Walking/Rolling tasks

PEOPLES-II moves using L-type and W-type. However, the right and left sides are controlled independently. Therefore, in order to avoid rolling and pitching a balanced control between both sides is needed. Also, we check whether one of each leg pair locates at upside on the ground because down-side candidate is impractical. Specifically, the case when two leg ends touch the ground at the same time should be avoided because the ground friction prevents the robot's sitting or standing. In the L-type, the robot is allowed to walk of constant γ , with short, standard or long strides (Okada et al., 2005). On the other hand, the legs have a tendency to direct upward so that all of the legs do not touch the ground in the W-type (Okada et al., 2006).

4.2 Switching task

In order to make the switch comfortably with less shock to the robot body, we separate the robot control into three phases to generate transitional adaptation in the switching task (Botelho et al., 2009). The steps taken by the robot are shown in Figure 4. The locomotion is switched from L-type to W-type from top to bottom. We divided the switching task in three phases. The first, second and last phases are the L-type, Switching and W-type, respectively. During the

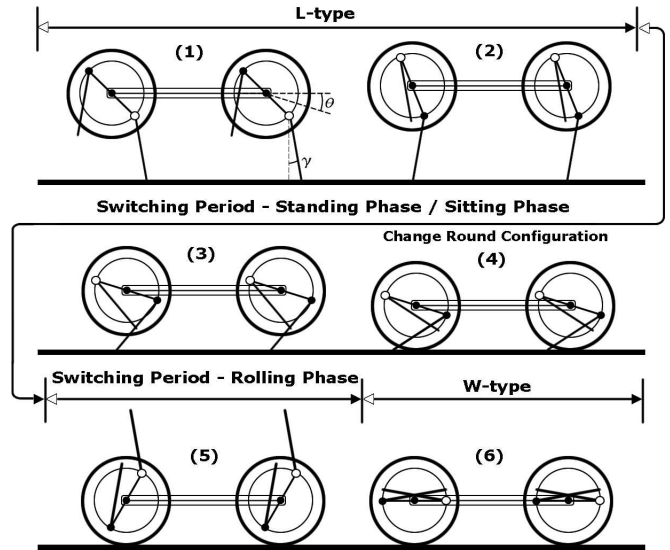


Figure 4: Motion sequence to make the switching task in a stable fashion.

L-type the value of γ is fixed, as shown in configurations (1)-(2). In the switching period the robot is able to stand or sit in configurations (3)-(4) and (4)-(3), respectively. In the sitting phase, the motions are continues until two landing contacts on the ground. It is called *change round configuration* (CRC). These landing happen when the robot stands from rolling or sits from walking as shown in (4). After that, transitional adaptation is needed in the rolling phase to make the connection between the rolling to the regular motion of W-type, as shown in configurations (5)-(6).

We propose three methodologies for doing preparatory tasks before starting the switching task (Okada et al., 2008). Each is characterized by θ and γ . The values of γ are calculated according to the rotation of θ in the switching task. In the first methodology, θ and γ are variables solving the 6th order Eq.6 and using Eq.3, respectively. It is the method for minimizing total energy cost explained in the Section 3. In the second methodology, θ is constant and γ is variable. The third methodology γ is constant and θ is variable. But, it is considered not practical because of the high energy consumption. In order to evaluate the energy cost for driving the two joint motors in the first methodology, we take into consideration the ratio of the energy cost for producing the J_h torque in comparison to the cost for producing the J_k torque. The values of θ and γ are uniquely calculated in order to reach the CRC. In this paper, we consider the first and second methodologies in the simulation and experimental verification.

Figure 3 shows the activity diagram with the algorithm uti-

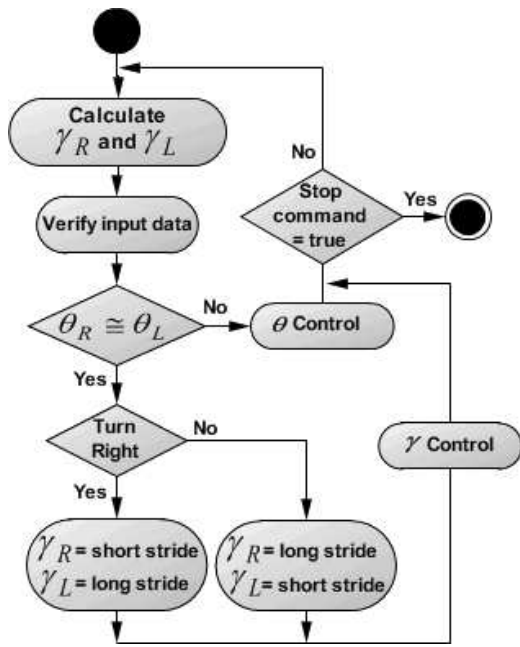


Figure 5: The activity diagram of the turning task.

lized to control the robot in the switching task. We use the notations: θ_t , θ_c and γ_t , γ_c for the target, current arm and leg posture angles, respectively. Also, δ , H_b , H_s and H_e are the pre-defined incremental constants, maximal H , minimal H and the indicator of the end hip joint height, sequentially. The first step is to assign H according to the switching task. For example, if the switch is from L-type to W-type (L→W) then H receives H_b and H_e which is equal to H_s in the sit down phase. In the reverse switch, H is equal to H_s . Also, H_e is same to H_b and H increases until reach the L-type. Next, the H and δ are assigned and θ_t and γ_t are calculated. The input data concerned with θ and γ are verified by potentiometers. If $\theta_c \cong \theta_t$ and $\gamma_c \cong \gamma_t$ then H increase or decrease and is assigned again to calculate the next θ_t and γ_t . After that, the γ control is verified. Using the same idea, for γ_c and γ_t . Next, the input data is verified. This idea is proceeded until H is equal to H_e .

4.3 Turning task

In the turning task, the amplitude of a stride and the rotation speed in the right/left sides are the main points to be considered for the L-type and W-type, respectively. Evidently, the legs are easily inclined and used to determine the amount of walking stride (Okada et al., 2006). That is, the distance between two leg ends located at the same J_h when the leg pair makes a landing together. The main idea is to control the amount of stride to make the robot walk forward or backward with short, standard and long strides (Okada

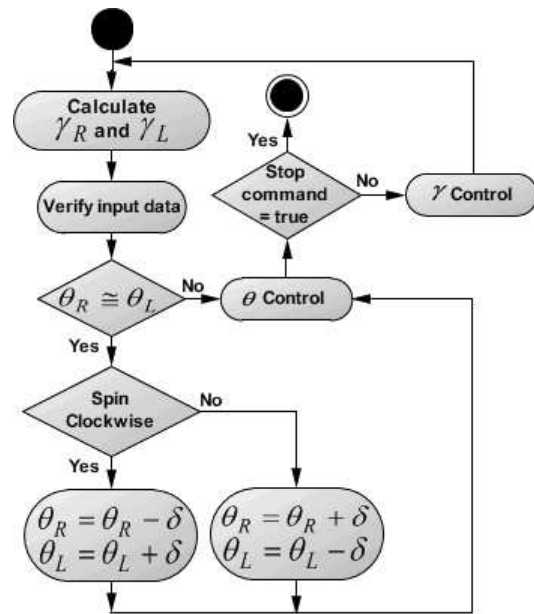


Figure 6: The activity diagram of the spinning task.

et al., 2004; Okada et al., 2005). Also, the ability of changing the amount of stride is important to make the robot walk over obstacles without bumping into them. The robot turns right by making a long stride to the left and a short to the right (Mahmoud et al., 2008). However, the turning task for the W-type is performed when the increment of θ on the right side is less than on the other side.

Figure 5 shows the activity diagram utilized in the turning by L-type. The amount of γ in the right (γ_R) and left (γ_L) sides of the robot is determined as a periodical function of θ (Okada et al., 2005). The current γ_c , θ_R on the right and other side θ_L are verified as input data. The synchronization is requested, therefore if $\theta_R \cong \theta_L$ and the task is to turn right then the values of γ_R and γ_L are short and long strides, respectively. Otherwise, the θ control explained in the Figure 3 is asked to make $\theta_R \cong \theta_L$.

4.4 Spinning task

The spinning task is performed when the robot turns around its body center. The θ in the right and left arms are driven in opposite directions in synchronization (Mahmoud et al., 2008).

In the spinning task, synchronization between right and left arms (notated as θ_R and θ_L) is required. However, they move in opposite directions. Therefore, if $\theta_R \cong \theta_L$ and the spin is performed clockwise then $\theta_R = \theta_R - \delta$ and $\theta_L = \theta_L + \delta$, as shown in Figure 6.

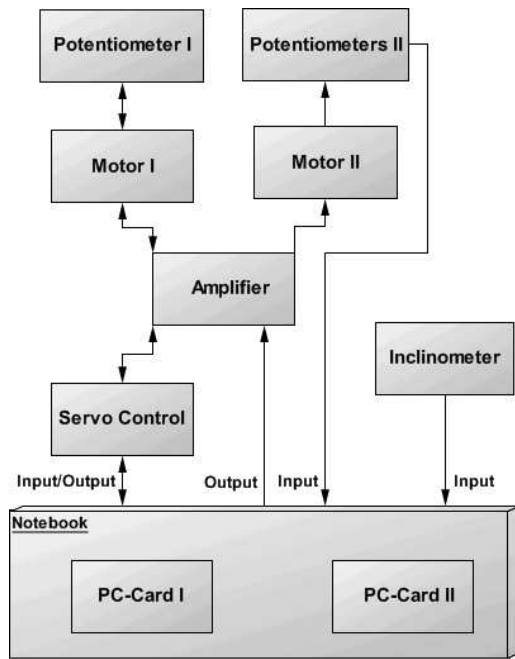


Figure 7: Diagram for hardware specifications.

5 CONTROLLER IMPLEMENTATION

This section presents an overview of the hardware and software control implementation of PEOPLER-II. The simulation and experimental interfaces with the class diagrams are explained. The communication between the hardware and software is also explained.

5.1 Hardware control specifications

Our hybrid robot is composed of a notebook with two PC-Cards, servo control, amplifier, motors, potentiometers and inclinometer. A diagram of the whole system is seen in Figure 7. The notebook and other components are represented by the cube and rectangular box, respectively.

The Athlon XP-M 1400 1.2GHz and 256 RAM with two Cardbus interfaces connect the two digital/analog I/O PC-Cards of ADA16-32/2(CB)F (CONTEC Co., Ltd., 2008). The bus mastering included in the Cardbus allows a controller on the bus to communicate with other devices without going through the CPU. It transfers analogue input, analogue output, digital input and digital output data and synchronizes them with clock signal. The eight legs and four J_h s are controlled by the PC-Card I and PC-Card II, respectively. The PC-Card I receives and sends the data through the servo control. However, the PC-Card II is directly connected to the amplifier and also receives the input data from the inclinome-

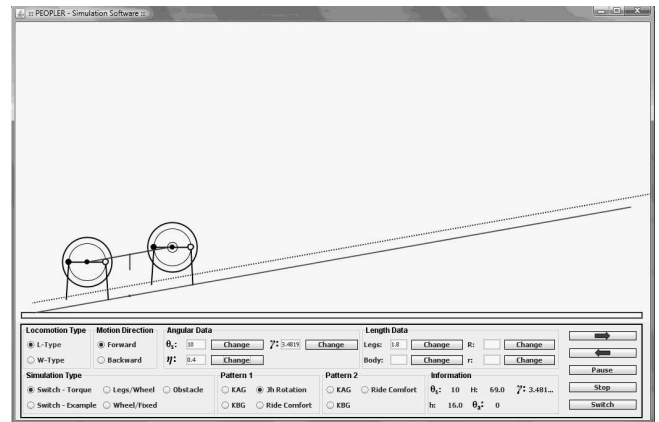


Figure 8: 2-D Simulation environment.

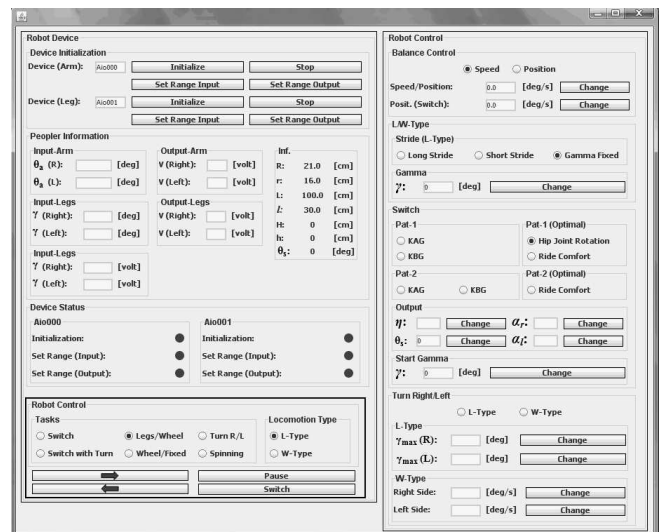


Figure 9: Experimental environment.

ter AccuStar SD-20 (Pacico).

The Motor I and Motor II on the right side of the robot are linked to three potentiometers, the first CPP-45 of $5k\Omega$ is utilized to get the current γ and the other two HP-16 of $10k\Omega$ are responsible for getting θ . The same number of motors and potentiometers are utilized to control the left side. These data are transmitted to the PC-Cards as an input data.

5.2 Software control design

The GUI of a computer program is the part that handles the output to the display and the input from the person using the program. The simulation and experimental interfaces are developed in JAVA to compose the software control.

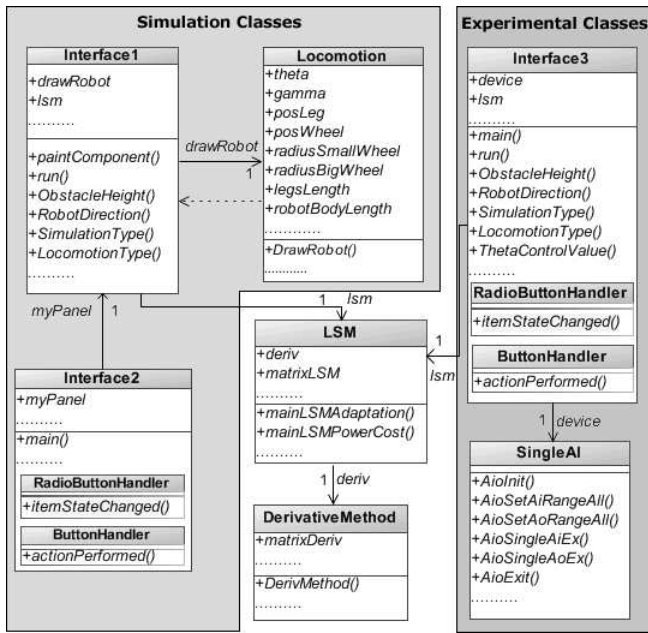


Figure 10: Class diagram for the PEOPLER-II application.

5.2.1 Simulation interface

Due to its symmetry on the right and left sides, and controlled in synchronization in motion of the front and rear structures, PEOPLER-II can be studied considering only the side view. Therefore, we developed a software in JAVA to simulate in 2-D the robot locomotion control, as shown in Figure 8. It provides a GUI that represents an environment in which we easily create, set and edit the configurations of our robot to perform the walking, rolling and switching tasks. The smoothness of the robot is also considered in our application. We tried to imitate the overall control of the robot with its real characteristics. The synchronization between the front and back, number of legs and wheels are also defined in the software. It is useful to imagine control scenarios without having to use the real prototype. On the bottom, several buttons are available for software interaction.

5.2.2 Experimental interface

Figure 9 shows the interface utilized to control the PEOPLER-II in the experiment. Some parameters are shown in the interface. For instance, the design notifications of R , r , L , l and the current input data of θ , γ , H .

5.2.3 Class diagrams

The class diagrams show an object oriented structure including attributes, their operations, and relationships to other

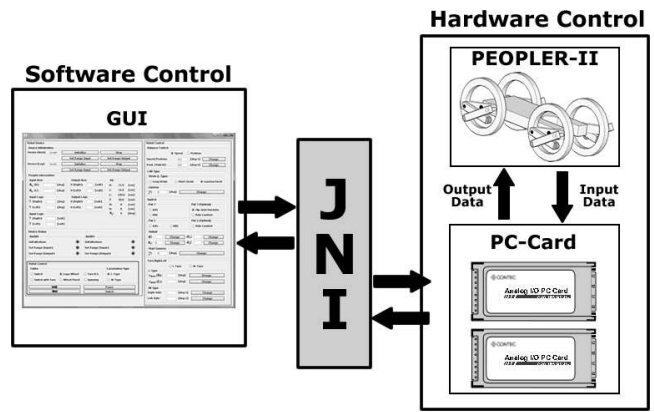


Figure 11: Communication between the software and hardware control.

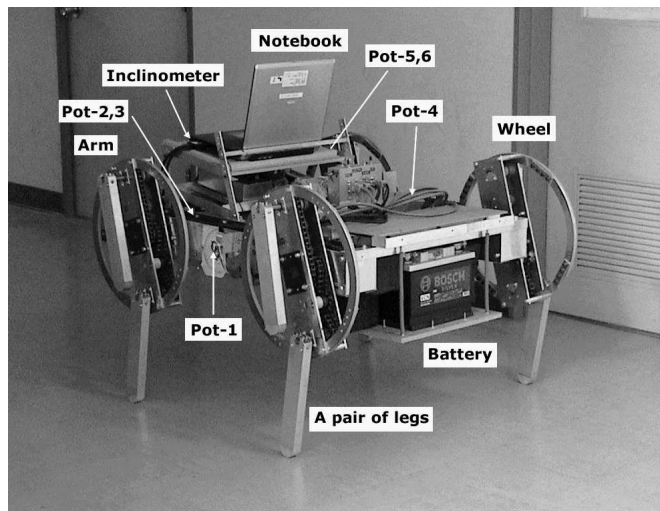


Figure 12: The prototype of the PEOPLER-II.

classes (Kim et al., 2006). Also, it describes the static view of an application (Purchase et al., 2001). The dependence between two classes is drawn as a line arrow characterized by a label showing the relation type, for instance the **Interface1** class is bound up together with **Locomotion** class because it contains an attribute of type *drawRobot* (Murnolo et al., 2001), as shown in Figure 10.

The application consists of five classes for simulation and four classes for experimental verification. However, two classes are designed to be reusable as components of other applications. These are the **LSM** (Least Square Method) and **DerivativeMethod**. Therefore, in total our software control program has seven classes. The class diagram lists the class name in bold. A description of each of the classes is shown below.

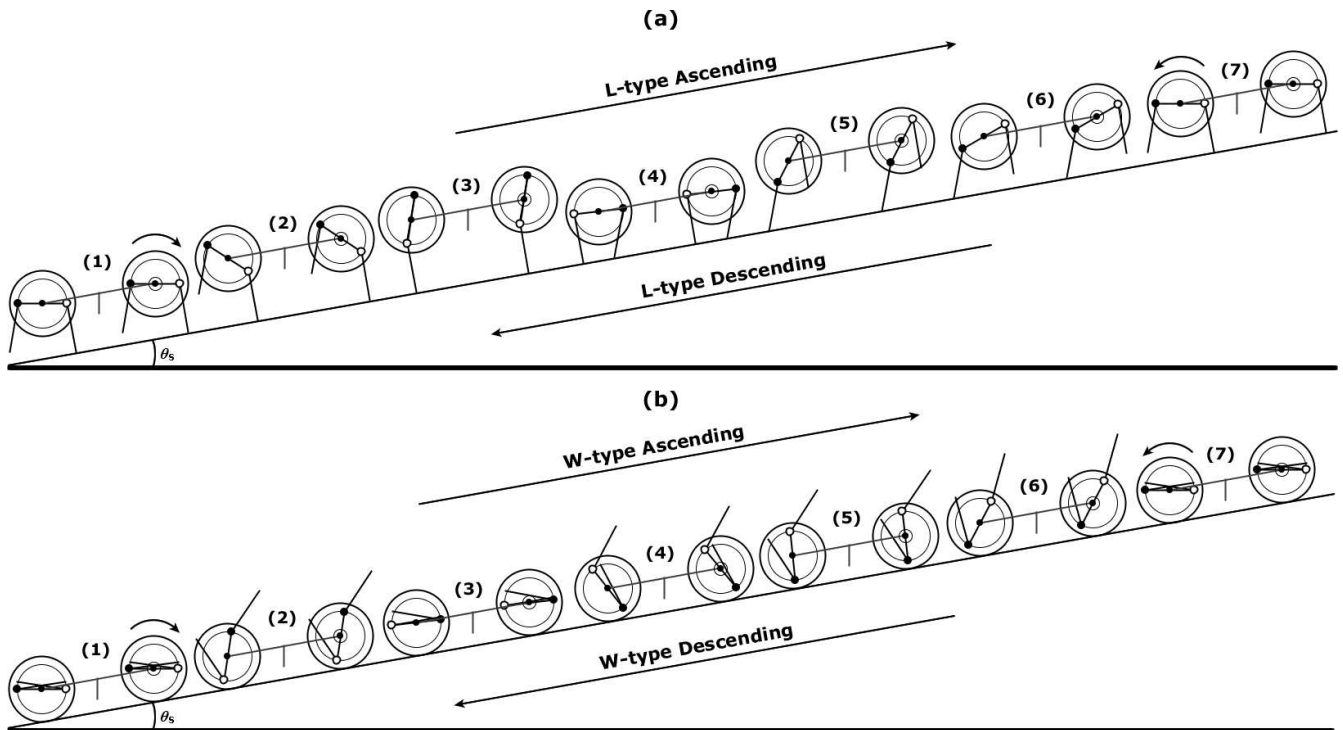


Figure 13: L-type with a fixed value of γ in (a) and W-type generated using Fourier function in (b) on the slope of $\theta_s = 10$ degrees.

Interface1 is responsible for drawing the side view of PEOPLER-II. Also, it receives all the information that the user is allowed to make changes with the aim to simulate the robot motion.

Interface2 creates an object of class **Interface1** and its necessary GUI.

Locomotion class receives robot specifications (R , r , l , θ and γ) to be generated on the coordinate system.

LSM is a class responsible for returning the estimated data calculated by the LSM (Okada et al., 2006).

DerivativeMethod is a class responsible for calculating θ in the 6^{th} order equation (Eq.6) in the method for minimizing total energy cost.

Interface3 is responsible for designing the experimental GUI.

SingleAI class is developed to use the driver library (API-PAC(W32)) (CONTEC Co., Ltd., 2008) supplied with the PC-Card.

5.3 Communication of hardware and software control

The structure of the system combines the user interaction through a software and hardware control. Basically, they are composed of the hardware specifications illustrated in Figure 7 and the GUI in Figure 9. In the hardware control, the PC-Card is the acquisition device utilized to make the connection between our GUI and the robot. The library API-PAC (W32) provides the commands for the device in the form of Windows standard Win32 API functions (DLL). It makes the high-speed application software easy to be created using programming languages that support Win32 API functions, such as Visual C++ .NET, Visual C# .NET, Visual Basic .NET, Delphi, C++ Builder, etc. However, the JAVA language is not supported by the device. Therefore, we use the framework called JAVA Native Interface (JNI) that allows JAVA code running in the JAVA virtual machine (JVM) to call and be called by native applications (programs specific to a hardware and operating system platform) and libraries written in other languages such as C, C++, Delphi, etc (Liang, 1999). The JNI servers as the glue between JAVA and native applications, as shown in Figure 11.



Figure 14: Experimental results in the walking, rolling using variable γ , rolling using fixed γ , switching from W-type to L-type, from L-type to W-type, turning by L-type, turning by W-type and spinning in (a), (b), (c), (d), (e), (f), (g) and (h), respectively.

6 SIMULATION AND EXPERIMENTAL RESULTS IN THE ROBOT TASKS

6.1 Prototype of PEOPLER-II

Figure 12 shows PEOPLER-II photograph. The 12 V battery and the notebook are shown in the figure. The potentiometers (Pot-1, Pot-4) in the middle of the robot body and (Pot-2,3, Pot-5,6) are responsible for getting input data of γ and θ ,

respectively. Also, the two-axis inclinometer is utilized to get the data θ_s .

6.2 Walking

Figure 13(a) shows the simulation results in the L-type using fixed value of $\gamma = 8$ degrees. The robot moves on an ascending and descending slope in the sequence (1)-(7) and (7)-(1), respectively. The experimental results shown in Figure 14(a)

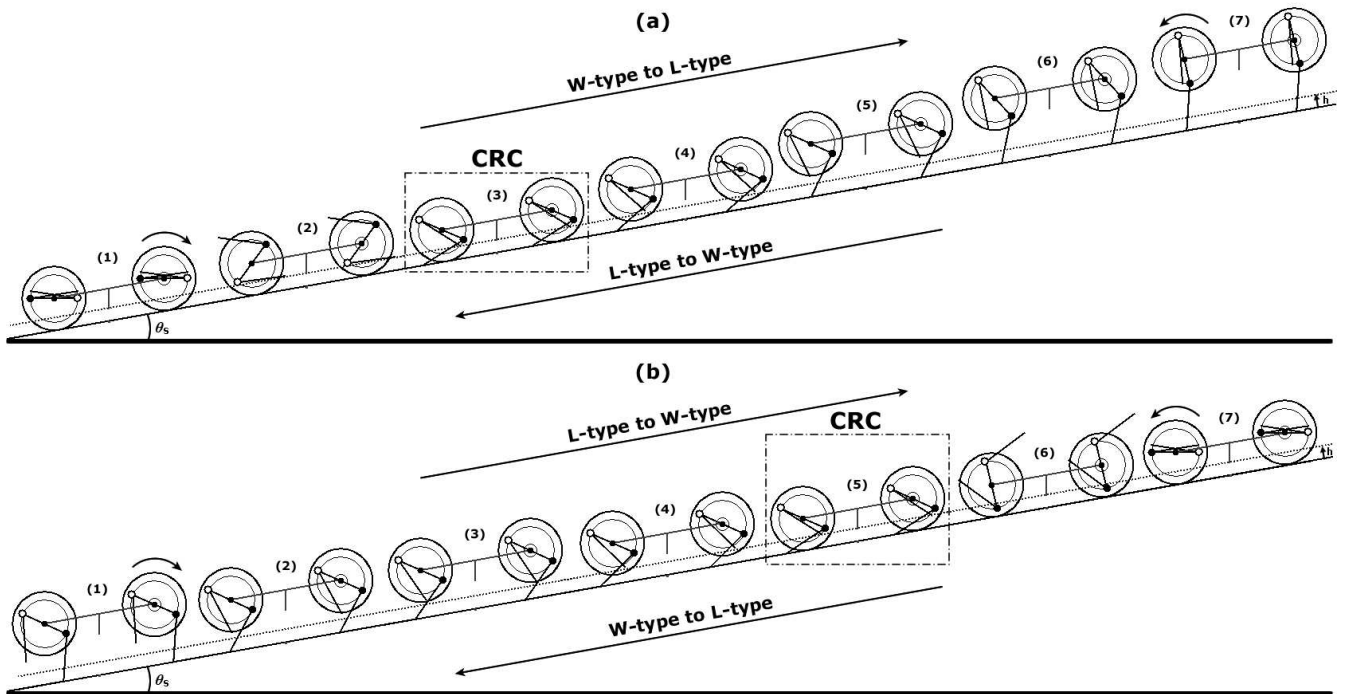


Figure 15: Optimal switching from W-type to L-type on an ascending slope and from L-type to W-type on a descending slope in (a). Optimal switching without arm motion from L-type to W-type and vice versa in (b) under $\theta_s = 10$ degrees.

confirmed the balanced control without rolling and pitching. It is clear to verify that the stride becomes long in the photos ①-②, ⑤-⑦ and short in ③-④ in Figures 13(a) and 14(a). These are typical characteristics when the constant value of γ is utilized.

6.3 Rolling

In the W-type, we obtained a faster, more stable and easy to control robot motion than L-type. Figures 13(b), 14(b) and 14(c) show the W-type using a variable γ calculated as a Fourier function of θ and γ is π radian, respectively. Configurations (1)-(7) in Figure 13(b) are the same as to photos ①-⑦ in Figure 14(b).

6.4 Switching without reverse rotation

The simulation results shown in Figure 15(a) are obtained by the first methodology. In this methodology, θ and γ are considered as variables and are solved by minimizing total energy cost explained in Section 3. The optimal switching from W-type to L-type on an ascending and from L-type to W-type on a descending slope under $\theta_s = 10$ degrees is performed in the same hip joint direction without a reverse rotation in the configurations (1)-(7) and (7)-(1), respectively.

Figure 14(d) shows the sequential experimental photos in the switching from W-type to L-type on an ascending slope. The inclinometer recognizes $\theta_s = 10$ degrees. It is performed by the activity diagram explained in Figure 3. The reverse motion from ⑦ to ① makes the switch from L-type to W-type on a descending slope. Photos from ③ to ⑦ correspond to the same sequence in the simulation shown in Figure 15(a). The photo ③ is the CRC.

6.5 Switching without arm motion

Figure 15(b) shows the simulation results using a fixed value of θ and variable γ . That is, the second methodology. Configurations (1)-(7) and (7)-(1) illustrate the switching from L-type to W-type on an ascending and vice versa on a descending slope, respectively. Because of the reversible sequence of motion when switching between W-type to L-type on an ascending and L-type to W-type on a descending, it can be understood from the configurations (5)-(1) and (1)-(5), respectively. The CRC is the configuration (5).

The experimental results concerned with the simulation in Figure 15(b) is illustrated in Figure 14(e). The photos ③, ④, ⑤, ⑥ and ⑦ are almost the same as to configurations (1), (2), (5), (6), (7) in Figure 15(b), respectively. The CRC

is seen in ⑤).

6.6 Turning and spinning

PEOPLER-II turns left by making a long stride right and a short stride left, as explained in Figure 5. Figure 14(f) shows the turning by L-type. However, Figure 14(g) shows the W-type turning right when the increment of θ on the right side is less than on the other side. Finally, using the control algorithm in Figure 6 the right and left arm motion is in opposite directions in synchronization in the spinning task, as shown in Figure 14(h).

7 CONCLUSIONS

In this paper we showed the adaptation of our legged robot PEOPLER-I to become a renovated hybrid robot PEOPLER-II. It is composed of 4DOFs; one arm set and one leg set for right and left sides. In total, it has only 4 motors, i.e. two motors for driving right and left hip joints and another two motors for rotating left and right legs. The hip joints and legs at front and back were mechanically synchronized.

The robot was able to walk, roll, switch, turn and spin. These were the robot tasks. The control algorithms of these tasks were explained. The control method for minimizing energy consumption was considered in the switching task. The hardware components and the connections between them were discussed. Also, we described the GUI utilized in the simulation and experiment. The structure of the system using the JNI framework was carried out in order to make the communication between the software and the robot. The simulation results in the walking, rolling and switching were showed. In the switching task, we took into considerations the same hip point direction and the constant state of the arm rotation. Also, we succeeded in demonstrating the robot tasks using the PEOPLER-II. These are the main goals achieved in this research.

Further work in the switching with turning and spinning is being considered to enhance the application of the robot. As we showed in the results that the robot is able to turn and spin, the switching with these tasks is also possible. If the leg pairs are controlled independently, the legs are able to be folded completely within the wheel frame. Therefore, the new design of the PEOPLER-II with 6DOFs is considered to reduce the vertical swing in L-type and to make the legs completely fold with the wheel frame in the W-type. In order to verify the vertical swing and the variation of the robot's inclination, the experimental trajectory of the robot center is considered in the next work. The PEOPLER-II is only a prototype for developing the hybrid robot. However, we believe that PEOPLER-II will be able to carry a human body.

Finally, we thank the Japanese Government Scholarship Program from the Ministry of Education, Culture, Sports, Science, and Technology (MONBUKAGAKUSHO) for the support of this research.

REFERENCES

- Aarne, H., Ilkka, L., Miso, M. e et al. (2001). Robot motion by simultaneously wheel and leg propulsion, *Proc. of 4th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Germany, pp. 1013–1019.
- Adachi, H., Koyachi, N., Arai, T. e et al. (1999). Mechanism and control of a leg-wheel hybrid mobile robot, *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, Korea, pp. 1792–1797.
- Akinfiyev, T., Fernandez, R. e Armada, M. (2008). Design of wheeled climbing robot with changeable structure, *Proc. of 11th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Portugal, pp. 549–554.
- Amaral, P. F. S., Gamah, H. E., Fontaine, J. G. e Rabit, J. (1993). RALPHY - um robô móvel eletropneumático com quatro pernas, *I Congresso Iberoamericano de Engenharia Mecânica*.
- Balasubramanian, R., Rizzi, A. A. e Mason, M. T. (2008). Legless locomotion a novel locomotion technique for legged robots, *J. of Robotics Research* 27(5): 575–594.
- Booch, G., Jacobson, I. e Rumbaugh, J. (1996). The unified modeling language for object-oriented development, *Documentation Set Version 0.91 Addendum UML Update - Rational Software Corporation, USA*.
- Borges, G. A., Lima, A. M. N. e Deep, G. S. (2003). Controladores cinemáticos de trajetória para robôs móveis com tração diferencial, *VI Simpósio Brasileiro de Automação Inteligente (SBAI 2003)*, pp. 787–792.
- Botelho, W. T., Okada, T. e Shimizu, T. (2009). Smooth switching phases control with simulation and experimental verification of a hybrid mobile robot from leg-type to wheel-type and vice versa, *Journal of Cybernetics and Systems, International Science Press, ISSN 0974-5459* 2(2): –.
- Browning, D. R., Trimble, J., Song, S. e et al. (1996). Legged mobility a wheelchair alternative, <http://www.ev1.uic.edu/drew/leggs.htm>.
- CONTEC Co., Ltd. (2008). url: <http://www.contec.com>.
- Dai, Y.-J., Nakano, E., Takahashi, T. e et al. (1995). Motion control of leg-wheel robot for an unexplored rough terrain environment, *Proc. of 7th Int. Conf. on Advanced Robotics*, Spain, pp. 911–916.

- Endo, G. e Hirose, S. (1999). Study on roller-walk (system integration and basic experiments), *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA)*, United States, pp. 2032–2037.
- Filho, A. B., Amaral, P. F. S., Pinto, B. G. M. e Lima, L. E. M. (2004). Uma metodologia para a localização aproximada de um robô quadrúpede, *XV Congresso Brasileiro de Automática*.
- Fu, Y. L., Li, Z., Yang, H. J. e et al. (2007). A wheeled wall-climbing robot with a climbing leg, *Proc. of 10th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Singapore, pp. 55–64.
- Gamann, B. e Berns, K. (2002). Local navigation of lauron iii walking in rough terrain, *Proc. of 5th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, France, pp. 509–514.
- Garcia, E. e de Santos, P. G. (2006). On the improvement of walking performance in natural environments by a compliant adaptive gait, *IEEE Trans. on Robotics* **22**(6): 1240–1253.
- Germann, D., Hiller, M. e Schramm, D. (2005). Design and control of the quadruped walking robot ALDURO, *Proc. of Int. Symp. on Automation and Robotics in Construction (ISARC)*, Italy, pp. 1–8.
- Grand, C., BenAmar, F., Plumet, F. e et al. (2004). Decoupled control of posture and trajectory of the hybrid wheel-legged robot Hylos, *Proc. of IEEE Int. Conf. on Robotics & Automation*, United States, pp. 5111–5116.
- Heinen, M. R. e Osório, F. S. (2007). Controle inteligente do caminhar de robôs móveis utilizando algoritmos genéticos e redes neurais artificiais, *XXVII Congresso da SBC - VI Encontro Nacional de Inteligência Artificial (ENIA)*, pp. 952–961.
- Junior, S. G. O. (2008). *Desenvolvimento de um robô com rodas autônomo*, Master's thesis, Universidade Federal do Rio de Janeiro - COPPE/UFRJ.
- Khoh, K. C. e Cho, H. S. (1999). A smooth path tracking algorithm for wheeled mobile robots with dynamic constraints, *J. of Intelligent and Robotic Systems* **24**(4): 367–385.
- Kim, B. e Tsiotras, P. (2002). Controllers for unicycle-type wheeled robots: Theoretical results and experimental validation, *IEEE Trans. on Robotics and Automation* **18**(3): 294–307.
- Kim, M., Kim, S., Choi, M. e et al. (2006). UML-based service robot software development: A case study, *Proc. of 28th Int. Conf. on Software Engineering*, ACM, China, pp. 534–543.
- Kimura, H., Fukuoka, Y. e Cohen, A. H. (2007). Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts, *J. of Robotics Research* **26**(5): 475–490.
- Konyev, M., Palis, F., Zavgorodnity, Y. e et al. (2008). Walking robot “ANTON”: Design, simulation, experiments, *Proc. of 11th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Portugal, pp. 922–929.
- Kühne, F., Claro, C. A. H., Suess, S. R. e Lages, W. F. (2004). Estimação e controle da posição de um robô móvel utilizando filtro de kalman descentralizado, *XV Congresso Brasileiro de Automática*.
- Lacagnina, M., Muscato, G. e Sinatra, R. (2003). Kinematics, dynamics and control of a hybrid robot wheeleg, *J. Robotics and Autonomous Systems* **45**(3): 161–180.
- Lages, W. F. e Hemerly, E. M. (1998). Controle em coordenadas polares de robôs móveis com rodas, *XII Congresso Brasileiro de Automática*.
- Lapierre, L., Zapata, R. e Lepinay, P. (2007). Combined path-following and obstacle avoidance control of a wheeled robot, *J. of Robotics Research* **26**(4): 361–375.
- Lauria, M., Piquet, Y. e Siegwart, R. (2002). OCTOPUS: an autonomous wheeled climbing robot, *Proc. of 5th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, France, pp. 315–322.
- Liang, S. (1999). The java native interface programmer's guide and specification, *Published by Addison-Wesley*.
- Lima, L. E. M., Amaral, P. F. S. e Filho, A. B. (2008). Controle da estabilidade de equilíbrio para um robô quadrúpede com uso de lógica nebulosa, *VIII Conferência Internacional de Aplicações Industriais - Induscon*.
- Mahmoud, A., Okada, T. e Shimizu, T. (2008). Circular trajectory simulation analysis of a rotating four-legged robot, *Proc. of Int. Conf. on Instrumentation, Control and Information Technology (SICE)*, Japan, pp. 691–694.
- Morales, R., Feliu, V., Gonzalez, A. e et al. (2004). Kinematics of a new staircase climbing wheelchair, *Proc. of 7th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Spain, pp. 249–264.
- Morales, R., Feliu, V., Gonzalez, A. e et al. (2006). Kinematic model of a new staircase climbing wheelchair and its experimental validation, *J. of Robotics Research* **25**(9): 825–841.

- Murnolo, E., Moratto, S., Nolich, M. e et al. (2001). Object oriented design of a mobile robot using UML, *Proc. of 23th Int. Conf. Information Technology Interfaces*, Croatia, pp. 608–613.
- Okada, T., Botelho, W. T. e Shimizu, T. (2006). Compatible use of a legged robot as a wheeled robot and its demonstrative simulation, *Proc. of 9th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Belgium, pp. 34–44.
- Okada, T., Botelho, W. T. e Shimizu, T. (2007). Leg control for changing locomotion between leg-type and wheel-type base on effective use of total power, *Proc. of 10th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Singapore, pp. 365–372.
- Okada, T., Botelho, W. T. e Shimizu, T. (2008). Walking gait control for making smooth locomotion mode change of a legged and wheeled robot, *Proc. of 11th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Portugal, pp. 711–719.
- Okada, T., Hirokawa, Y. e Sakai, T. (2003). Development of a rotating four-legged robot, PEOPLER for walking on irregular terrain, *Proc. of 6th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Italy, pp. 593–600.
- Okada, T., Hirokawa, Y., Sakai, T. e Shibuya, K. (2004). Synchronous landing control of a rotating 4-legged robot, PEOPLER, for stable direction change, *Proc. of 7th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, Spain, pp. 85–96.
- Okada, T., Sakai, T., Shibuya, K. e Shimizu, T. (2005). Hip joint control of a legged robot for walking uniformly and the self-lock mechanism for compensating torque caused by weight, *Proc. of 8th Int. Conf. on Climbing and Walking Robots (CLAWAR)*, United Kingdom, pp. 197–204.
- Palis, F. e Rusin, V. (2004). Adaptive impedance control of legged robot, *Proc. of 11th Int. Conf. on Power Electronics and Motion Control*, Latvia.
- Pizziolo, T. A., de S. Martins-Filho, L., de Resende, P. e Silvino, J. L. (2004). Análise de simplificação na modelagem dinâmica aplicada a robôs quadrúpedes, *SBA Controle & Automação* **15**(3): 301–308.
- Poulakakis, I., Papadopoulos, E. e Buehler, M. (2006). On the stability of the passive dynamics of quadrupedal running with a bounding gait, *J. of Robotics Research* **25**(7): 669–687.
- Purchase, H. C., Colpoys, L., McGill, M. e et al. (2001). UML class diagram syntax: An empirical study of comprehension, *Australian Symposium on Information Visualisation*, Australia, pp. 113–120.
- Richter, W. M. (2001). The effect of seat position on manual wheelchair propulsion biomechanics: a quasi-static mode-based approach, *Medical Engineering and Physics* **23**: 707–712.
- Schammas, A., Caurin, G. A. P. e Valente, C. M. O. (2001). Control of a one-legged robot with energy savings, *J. of the Brazilian Society of Mechanical Sciences* **23**(1): 41–48.
- Schroueder, G. N., Espíndola, D. B., Botelho, S. S. C., Bicho, A. L. e Oliveira, V. M. (2005). Simulador gráfico para controle de robôs móveis omnidirecionais, *INFOCOMP - J. of Computer Science* **4**(4): 38–47.
- Seelinger, M., Yoder, J., Baumgartner, E. T. e et al. (2002). High-precision visual control of mobile manipulator, *IEEE Trans. on Robotics and Automation* **18**(6): 957–965.
- Sonehara, K., Yamamoto, T., Koichi, T. e et al. (2004). Development of breadboard model for locomotion of 3-leg wheeled type robot, *Tech. Rep. of IHI*.
- Still, S., Hepp, K. e Douglas, R. J. (2006). Neuromorphic walking gait control, *IEEE Trans. on Neural Networks* **17**(2): 496–508.
- Woo, C. K., Choi, H. D., Yoon, S. e et al. (2007). Optimal design of a new wheeled mobile robot based on a kinetic analysis of the stair climbing states, *J. of Intelligent and Robot Systems* **49**(4): 325–354.