

MASTER'S THESIS

L

Control System for a Spherical Robot

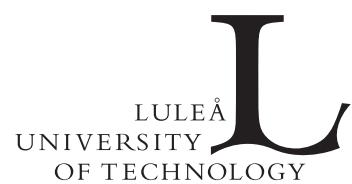
Masaki Nagai

Luleå University of Technology

Master Thesis, Continuation Courses
Space Science and Technology
Department of Space Science, Kiruna



TEKNILLINEN KORKEAKOULU
Faculty of Electronics, Communications and Automation
Department of Automation and Systems Technology



Masaki Nagai

Control System for a Spherical Robot

Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Technology

Espoo August 8, 2008

Supervisors:

Professor Aarne Halme

Helsinki University of Technology

Professor Kalevi Hyyppä

Luleå University of Technology

Instructor:

Tomi Ylikorpi

Helsinki University of Technology

Acknowledgements

First of all, I would like to acknowledge Professor Aarne Halme, Automation and Systems Technology Department, TKK, Finland, for giving me this opportunity to survey and study the field of spherical robots as well as giving me important comments on my work. Also, I wish to acknowledge Professor Kalevi Hyyppä, Department of Computer Science and Electrical Engineering, Luleå University of Technology, Sweden, for his constructive comments given at every important milestone of this research.

I am particularly grateful to Mr. Tomi Ylikorpi, Automation and Systems Technology Department, TKK, for guiding me to the completion of my thesis and taking care of every single aspect throughout the project, including procurement, paper work, designing the mechanical part of the robot, as well as managing the SpaceMaster Programme as a whole.

I would also like to gratefully appreciate the language support offered by Mr. William Martin, Department of Electrical Engineering, TKK. His support helped me greatly to improve the understandability and structure of the thesis. Needless to say, he helped me with the language correction.

I would also like to thank Mr. Tapani Leppänen, Automations and Systems Technology Department, TKK, for supporting my thesis with manufacturing the mechanical part of the prototype robot.

Furthermore, we must remember that Ms. Anja Hänninen, the Study Planner and Administrative coordinator of SpaceMaster Program, TKK, has assisted the SpaceMaster programme in administrative duties.

Finally, I would like to extend my heartfelt gratitude to my student colleagues for their understanding, assistance, and encouragement: Mr. Tran Duy Vu Bui (Richard), Mr. Felix Cabrera Garcia, Mr. Khurram Gulzar, Mr. Jan Philipp Hakenberg, Mr. Timo Paavo Kalevi Heiskanen, Mr. Farrukh Iqbal Sheikh, Mr. Tawon Uthaicharoenpong (Pipe), and Miss Vicky Yuen Kwan Wong.

Espoo, August 8, 2008

Masaki Nagai

永井 将貴

Author:	Masaki Nagai
Title of the thesis:	Control System for a Spherical Robot
Date:	August 8, 2008
	Number of pages: 71
Faculty:	Faculty of Electronics, Communications and Automation
Department:	Automation and Systems Technology
Program:	Master's Degree Programme in Space Science and Technology
Professorship:	Automation Technology (Aut-84)
Supervisors:	Professor Aarne Halme (TKK) Professor Kalevi Hyypä (LTU)
Instructor:	Tomi Ylikorpi
<p>Spherical robots utilize an original locomotion system by displacing its center of gravity in order to generate torque and rotate itself. They are believed to have several benefits. First, the only contact point with the ground allows locomotion with minimal friction, leading to low-energy motion. Second, the spherical structure allows the motion in tightly constrained spaces. Third, the spherical body protects the inner structure against external shocks or dust. Finally, it cannot lose mobility simply by overturning thanks to its spherical shape.</p> <p>Spherical robots, however, can generate only a small amount of torque for rotation. This makes its locomotion ability for up-hill climbing and obstacle overrunning greatly limited. Therefore, this study focuses on spherical robots operating on smooth flat floors. It describes the basic dynamics and control methods, which can still be applied for indoor use. This is an important step for further investigation of more complex dynamics and control methods.</p> <p>First, the dynamics of a spherical robot is investigated for two motions: driving and steering. Second, this paper describes the newly developed prototype robot, including its wireless communication system. The wireless system is one of the critical components of the prototype, since rotation does not allow physical wiring. Third, in order to avoid clumsy and unstable motion coming from the spherical shape, several control methods for smooth locomotion have been developed using the prototype.</p>	
Keywords:	pendulum, Bluetooth, PID control, ball robot, spherical robot

Contents

1	Introduction	1
2	Literature Review	3
2.1	History of Ball Robots	3
2.2	Principles of Locomotion	5
2.2.1	Using an IDU (<i>Inside Driving Unit</i>)	5
2.2.2	Using a Pendulum	7
2.2.3	Deformable Body	9
2.2.4	<i>Gyrover</i>	11
2.3	Dynamics Analysis	12
2.3.1	Analysis of an IDU-driven Robot	12
2.3.2	Analysis of pendulum driven robot	16
2.4	Applications	16
2.4.1	Playing Toy	16
2.4.2	Surveillance	17
2.4.3	Reconnaissance	18
2.4.4	Space Exploration	19
2.4.5	Entertainment	20
3	Theory	21
3.1	Dynamics of Driving	21
3.1.1	Horizontal Motion	21
3.1.2	Uphill and Downhill Motion	23
3.1.3	Obstacle crossing	24
3.2	Dynamics of Steering	26
3.2.1	Modeling	26
3.2.2	Ball traveling at low speed	27
3.2.3	Ball traveling at intermediate speed	28
3.2.4	Ball traveling at high speed	30

3.2.5	Numerical Calculation	32
4	Prototype Robot	33
4.1	System Overview	33
4.2	Electronic Architecture	34
4.2.1	Block Diagram Description	35
4.2.2	Magnetic Compass	36
4.2.3	Tri-axial Accelerometer	36
4.2.4	Gyroscope	37
4.2.5	Servo Motor	37
4.2.6	DC Motor	38
4.2.7	Battery	39
4.2.8	Motor Controller	39
4.2.9	Optical encoder	40
4.2.10	Main board	41
4.2.11	Main switch	41
4.2.12	Mounting	41
4.3	Specifications	42
5	Control Methods: Driving	44
5.1	Background	44
5.2	Stability of the robot rotation	45
5.2.1	Voltage to Ball Rotation	46
5.2.2	Motor Speed to Ball Rotation	49
5.2.3	Motor Torque to Ball Rotation	49
5.3	Control method	50
5.3.1	Rotation Speed Control	50
5.3.2	Torque Control	52
5.3.3	Other assistant controls	52
6	Control methods: Steering	54
6.1	Introduction	54
6.2	Stabilizing the roll angle	55
6.2.1	Modeling	55
6.2.2	Determination of roll angle	57
6.2.3	Stabilization Method	58

7 Simulation and Test Results	59
7.1 Driving Control	59
7.2 Roll angle stabilization	61
7.3 Driving angle limitation	62
7.4 Steering characteristics	62
8 Conclusions	64
8.1 Prototype robot	64
8.1.1 Bluetooth communication	64
8.1.2 Low self-discharge Ni-MH battery	65
8.2 Sensors for the feedback control	65
8.3 Theoretical analysis	65
8.3.1 Stability without sensors on the ball	65
8.3.2 Gyro effect	66
8.4 Control methods	66
8.4.1 Stability	66
8.4.2 Driving control	66
8.4.3 Steering control	67
8.4.4 Future works	68
References	69
Appendices	72
A Main Board Description	72
B Battery Charging	79
C Prototype Robot Operation	80
C.1 Introduction	80
C.1.1 Definition, Benefits, and Purpose of the Procedure	80
C.1.2 Intended Audience	81
C.1.3 Prior Knowledge and Skills Needed by the Audience	81
C.1.4 Brief Overall Description of the Procedure	81
C.1.5 Materials, Equipment, and Special Conditions	81
C.1.6 Working Definitions	82
C.1.7 Warnings, Cautions, Dangers	82
C.1.8 List of Major Procedures	82

C.2	Required Steps	83
C.2.1	Switching on <i>MarsBall</i>	83
C.2.2	Starting <i>MarsBallController</i>	83
C.2.3	Connecting to <i>MarsBall</i> from your computer	83
C.2.4	Controlling the <i>MarsBall</i> with the <i>MarsBallController</i>	84
C.2.5	Connecting the Joystick (Option)	85
C.2.6	Controlling <i>MarsBall</i>	86
C.2.7	Data Acquisition	87
C.2.8	Terminating Control of <i>MarsBall</i>	88
C.2.9	Disconnecting <i>MarsBall</i>	88
C.2.10	Shutting down the <i>MarsBall</i>	89
D	Discussion about Rotational Radius	90
E	Friction Evaluation	92
E.1	Static friction coefficient	92
E.2	Friction for driving	93
E.3	Friction for steering	93

List of Figures

2.1	<i>Mechanical Toy by E. E. Cecil (U.S. Patent 933,623)</i>	4
2.2	<i>The hamster ball design by A.D. McFaul (U.S. Patent 1,263,262)</i>	4
2.3	<i>IDU of the Rollo robot 2nd generation developed at TKK (reproduced from Spitzmueller, 1998)</i>	5
2.4	<i>IDU of the Rollo robot 3nd generation developed at TKK (Copied from the Automation and Systems Technology Department, TKK)</i>	6
2.5	<i>The Squiggleball (reproduced from Ylikorpi, 2005)</i>	7
2.6	<i>The Mars Ball (Ylikorpi, 2005)</i>	8
2.7	<i>The IWBS Robot (reproduced from Koshiyama 1992a)</i>	9
2.8	<i>The "Tumbleweed Ball"(reproduced from Jones 2001)</i>	10
2.9	<i>Soft body spherical robot Koharo (copied from the Koharo project website at Ritsumeikan University, Japan, http://www.ritsumei.ac.jp/se/hirai/research/softrobot-j.html)</i>	10
2.10	<i>Gyrover (copied from the Gyrover project website, http://www.cs.cmu.edu/cyber scout/gyrover.html)</i>	11
2.11	<i>Basic Principle for Turning</i>	12
2.12	<i>Modeling (reproduced from Halme, 2005a)</i>	13
2.13	<i>Steering Motion Modeling (reproduced from Koshiyama, 1996)</i> . .	14
2.14	<i>Roball(left), A boy playing with a Roball(right)(both reproduced from Michaud 2000)</i>	17
2.15	<i>Surveillance Robot (reproduced from Knight, 2005)</i>	17
2.16	<i>Cyclops (reproduced from Chemel et al., 1999)</i>	18
2.17	<i>Conical Mass (reproduced from Chemel et al., 1999)</i>	19
2.18	<i>Spherical Micro Robots sent to Mars (reproduced from Young, 2006)</i> . .	20
2.19	<i>Entertainment Robot Q.taro (reproduced from Nakamura 2002)</i> . .	20
3.1	<i>Modeling of the ball (driving)</i>	22
3.2	<i>Modeling of the ball (uphill/downhill)</i>	24
3.3	<i>Obstacle crossing</i>	25

3.4	<i>Modeling for steering</i>	26
3.5	<i>Modeling for steering with low speed</i>	27
3.6	<i>Modeling for steering with intermediate speed</i>	28
3.7	<i>Forces on each element</i>	28
3.8	<i>Rotational radius against pendulum steering angle</i>	32
4.1	<i>Prototype robot</i>	33
4.2	<i>System Overview</i>	34
4.3	<i>Block Diagram</i>	34
4.4	<i>The Crumb128-CAN Module (left), The IOGEAR Bluetooth Serial Adapter (right)</i>	35
4.5	<i>The Hitachi HM55B Compass Module</i>	36
4.6	<i>The position of the compass</i>	37
4.7	<i>The Kionix Tri-axial Accelerometer KXM52-1050 (left), The Kon-do KRG-3 Gyro (right)</i>	38
4.8	<i>The BlueBird BMS660MG-HS servo motor (made in Taiwan)(left), The Mabuchi RS540SH DC Motor (right)</i>	39
4.9	<i>Sanyo Low self-discharge NiMH battery Eneloop (left), 6-AA-battery pack used for the prototype (right)</i>	39
4.10	<i>The standard motor controller used at TKK (left), The handmade optical encoder with a photo coupler (right)</i>	40
4.11	<i>The main board</i>	41
4.12	<i>The main switch (left), The holder of the electronics (right)</i>	42
4.13	<i>The whole electronic system(left: front view, right: rear view)</i>	42
5.1	<i>Block diagram of driving motion Stabilization</i>	52
6.1	<i>Position of accelerometer</i>	57
6.2	<i>Block Diagram of Sideways Motion Stabilization</i>	58
7.1	<i>Simulation result of the motor speed control</i>	59
7.2	<i>Test result of motor speed control</i>	60
7.3	<i>Simulation result of roll angle stabilization</i>	61
7.4	<i>Test result of roll angle stabilization</i>	62
7.5	<i>Test results of steering</i>	63
7.6	<i>Prototype robot and its trajectory</i>	63
8.1	<i>Block diagram of the ball robot</i>	67

A.1	<i>Circuit design of the main board</i>	72
A.2	<i>Main Board Layout</i>	73
A.3	<i>Circuit design of the motor controller</i>	77
A.4	<i>Motor controller trax layout</i>	78
A.5	<i>Motor controller assembly drawing</i>	78
B.1	<i>SANYO Eneloop and its propriety charger</i>	79
C.1	<i>MarsBallController Icon</i>	83
C.2	<i>MarsBallController window</i>	84
C.3	<i>Communication Groupbox (Choose COM port and click Connect)</i>	85
C.4	<i>Communication Groupbox after successful connection with the green textbox</i>	85
C.5	<i>Joystick Groupbox</i>	86
C.6	<i>Control Groupbox (Click Start first, then drag ticks in the trackbars)(Left side: before start controlling, Right side: after start controlling</i>	86
C.7	<i>Acquired data in graphs</i>	87
C.8	<i>Received Data and Transmitted Command</i>	88
C.9	<i>Control Groupbox (Click "Stop"to stop operation of MarsBall)(Left: before stop controlling, Right: after stop controlling</i>	88
C.10	<i>Communication Groupbox (Click Disconnect to disconnect and close the COM port)(Left: before disconnection, Right: after disconnection</i>	89
D.1	<i>Spherical robot (left), closeup of between the robot and the surface (middle), ground track of the robot(right)</i>	90
E.1	<i>Simulation model for determining rotation acceleration of the robot</i>	93
E.2	<i>Friction force needed for steering</i>	93

List of Tables

4.1	Specifications of the pendulum	42
4.2	Specifications of the ball	43
4.3	Specifications of the DC motor	43
4.4	Specifications of the battery	43
4.5	Specifications of the electronics	43
A.1	Component location on the main board	74
A.2	Component list	75
A.3	Software list	76
E.1	Friction coefficients (static)	92

Symbols and Abbreviations

IDU	Inside Driving Unit
TKK	Teknillinen korkeakoulu (Helsinki University of Technology)
AS	Automation and Systems Technology Department at TKK
JPL	Jet Propulsion Laboratory (NASA Research Center in California, USA)
IWBS	Inside Wheel Balancing System
MIT	Massachusetts Institute of Technology
Q.taro	Quasi-stable Traveling and Action Robot
CG	Center of Gravity
OBC	On Board Computer
GS	Ground Station
AVR	Chip microcontroller developed by the Atmel Corporation
DC	Direct Current
PID	Proportional-Integral-Derivative
LSD	Low Self-Discharge
NiMH	Nickel-Metal Hydride
USART	Universal Synchronous Asynchronous Receiver Transmitter
LAN	Local Area Network
ADC	Analog-to-Digital Converter
RS232	Recommended Standard 232
USB	Universal Serial Bus
CAN	Controller Area Network
DIP	Dual Inline package
AA	AA battery type
LED	Light-Emitting Diode
PWM	Pulse Width Modulation

Chapter 1

Introduction

Among diverse types of mobile robots, spherical robots have become increasingly attractive in the last decade. They are believed to have several benefits. First, they have only a single contact point with the ground with minimal friction for locomotion. Therefore, they would be able to save energy for locomotion. Second, the spherical structure makes it possible to move even in tightly constrained spaces. Third, the spherical exoskeleton can protect the inner structure against external shocks or dust. It is even possible to be made liquid and gas proof. Finally, there is no chance for a ball to tip over, turn over, and then lose mobility. This is critical for other types of robots, such as humanoid robots and rover robots.

Until now, a considerable amount of research has been conducted on realizing spherical robots. Jones (2001) proposed a wind-driven spherical robot for planetary exploration on Mars. It is a large inflatable ball with scientific instruments inside its 6m-diameter body. The ball is controlled with a simple strategy – it is stopped by partial deflation and be driven again by full inflation. The idea of the wind-driven robots has been inherited to another spherical robot called the "*Mars Ball*". (Ylikorpi, 2005) It has a two-degree-of-freedom pendulum inside for active locomotion as well as fins harnessing the wind on Mars for passive locomotion.

Except for this wind-driven type of spherical robots, they can normally generate only little amount of torque to rotate themselves. This makes it difficult for the robots to overcome obstacles and climb uphill. Therefore, this study focuses on

spherical robots operating on smooth flat floors for indoor use. Some previous studies also discuss the locomotion on flat floors as follows.

Koshiyama (1996) has analyzed the dynamics of spherical robots and developed a prototype robot. First, he separated driving and steering motions for simplicity. Then Lagrange equations are used to obtain motion equations. He also conducted an experiment of the motion using the prototype robot. Laplante (2005) offered a mathematical background and simulation results determining the relationship between the steepness of steering and the amount of displacement of the inner mass of a spherical robot. As expected, slower speeds and larger steering angles lead to a steep steering of the robot. Spitzmueller (1998) has developed a prototype and invented a simple open-loop control method for it.

However, there are some problems in previous studies. First, the dynamics analysis has ignored the possible gyro effect of a rolling spherical body. The gyro effect derives from the rotation of the spherical body during the movement of a spherical robot. Although this effect can be disregarded when the speed is slow, it cannot be neglected when the speed is high.

Previous studies have also failed to develop control methods supporting manual control. So far, manual control has been very clumsy in previous studies, even with feedback control of a human operator. Feedback in this context means sensing the status of a robot with human eyes and controlling the robot accordingly.

Hence, the aim of this research is to develop a dynamic model as well as suitable control methods for supporting the manual control of the flat-floor operation. This study focuses on (1) developing the dynamics model, especially for turning that takes into account the gyro effect, (2) developing a prototype robot, and (3) proposing control methods for supporting manual control.

The rest of the paper is organized in the following manner. Chapter 2 reviews the history and previous studies of spherical robots. Chapter 3 shows the basic theory of its dynamics. Chapter 4 describes the developed prototype robot. Chapter 5 proposes a control method for driving. Chapter 6 proposes control methods for steering. Chapter 7 shows test results of the control methods using the prototype. Chapter 8 contains concluding remarks based upon the results of the tests.

Chapter 2

Literature Review

This study reviewed the literature in several stages in order to improve our understanding of the current status of spherical-robot research.

2.1 History of Ball Robots

A short history of self-propelled movable balls was outlined by Ylikorpi (2005) in his paper entitled "A Biologically inspired rolling robot for planetary surface exploration". The first movable ball was a spring-propelled toy patented in 1893 (U.S. Patent 508,558), which rolls on a straight path. The design was improved by B. Shorthouse in the year 1906 to make the ball follow a predefined curved trajectory. Since then, various mechanisms have been proposed and patented, thus allowing for the irregular motion of self-propelled movable balls. Figure 2.1 shows one design invented in 1909 for an amusement toy. It has a pendulum inside, which creates motion of the ball. The invention of the locomotion system of displacing the center of mass, which is currently still being used in most spherical robots, dates back a century.

Another popular locomotion system was patented by A.D. McFaul in 1918. It is known as the *hamster-ball* design. This design utilizes a wheeled counterweight in a ball. Today, this counterweight is known as an IDU, or *Inside Driving Unit*, and is currently employed in some spherical-robot designs. The main feature of this

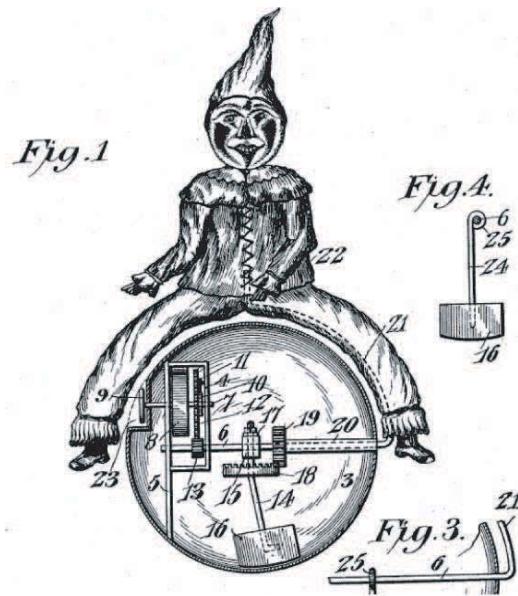


Figure 2.1: *Mechanical Toy by E. E. Cecil (U.S. Patent 933,623)*

design is that rolling motion comes from the friction between the inner surface of the spherical body and the IDU wheels. This requires small torque for a propelling system, unlike the pendulum locomotion system mentioned above, which requires a significantly greater amount of torque.

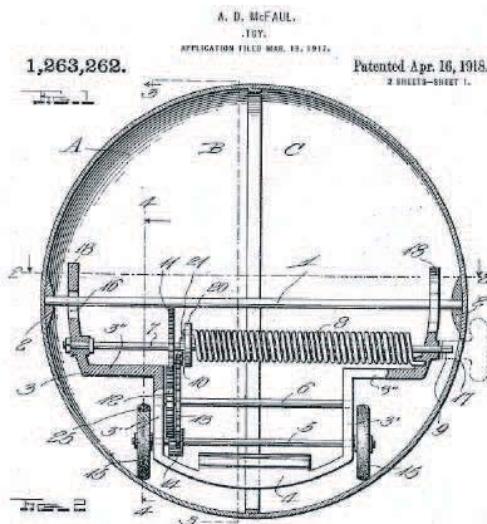


Figure 2.2: *The hamster ball design by A.D. McFaul (U.S. Patent 1,263,262)*

Subsequently, the use of a spring, as the power source, was replaced with batteries and electric motors in the patent filed by J.M. Easterling in 1957. The addition of an electric system to the spherical robots brought lots of functions for the robot,

such as sensing and switching.

2.2 Principles of Locomotion

Although most spherical robots utilize displacement of the center of mass as their driving force, they can vary in the strategies used for displacing the mass. This section discusses three locomotion principles.

2.2.1 Using an IDU (*Inside Driving Unit*)

One locomotion method is to use an IDU, or a box with wheels, which moves on the inner surface of the spherical body. One example of an IDU-driven spherical robot is the *Rollo* robot 2nd generation developed at TKK in 1998.(Spitzmueller, 1998) Its IDU has two driving wheels sideways and two un-driven wheels on the top to ensure a tight fit in the sphere. This robot employed electronics, such as sensors, motors, a transceiver, and a computer. Those components are necessary to ensure the smooth control of the robot. Dynamics modeling and rough open-loop control were conducted and the control turned out to be sufficiently accurate.

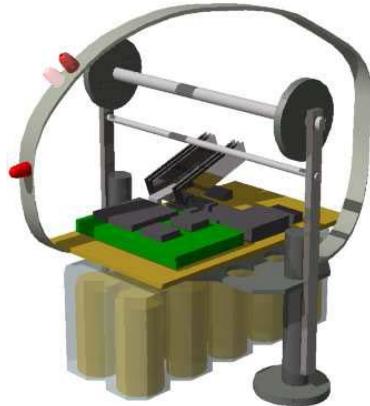


Figure 2.3: *IDU of the Rollo robot 2nd generation developed at TKK (reproduced from Spitzmueller, 1998)*

It has encoders, a gyro, and an inclinometer. Since the locomotion system utilizes an IDU, two encoders for both wheels are equipped. Without any reference pose information, the accuracy of the data has not been evaluated. Instead, the data

was just used for estimating rough dynamic motions, such as the vibration, of the robot. The sensor data is not even used for the control.

The 3rd generation *Rollo* following the 2nd generation has the different design of an IDU. Its IDU can turn itself by rotating along the rim gear on the inner surface of the ball, thus selecting the orientation of the ball on the spot. The driving motion is generated by swinging up the main mass of the IDU. This driving strategy is same as that of pendulum-driven type spherical robots mentioned in the next section. This study, however, classifies *Rollo* as an *IDU-driven* robot.



Figure 2.4: *IDU of the Rollo robot 3nd generation developed at TKK (Copied from the Automation and Systems Technology Department, TKK)*

One limitation, however, of IDU-driven robots is their size constraint. As a matter of fact, this type of robot requires a smooth and hard spherical body in order to let the IDU drive on the inner surface. Hard spherical bodies can easily be employed for small robots, such as the robots mentioned above. Furthermore, large plastic hard bodies require thick shells for structural strength, which leads to significant increases in weight. This could be critical in space missions where prohibitively high launch costs greatly depend on the weight of a payload. This creates something of a great paradox, since the larger the spherical robot is, the easier it is for it to overcome rough and rocky terrain on Mars and the Moon.

2.2.2 Using a Pendulum

Another way to displace mass for locomotion is to utilize a pendulum. Pendulums are usually connected to the geometric center of the sphere, without touching the exoskeleton. Therefore, any type of spherical body can be employed, such as inflatable, deployable, or wired bodies.

The *Squiggle Ball* toy

One example of such robots is the *Squiggle Ball* toy, distributed by Hart Toys, Inc. (<http://www.harttoys.com>) It has a driving mass called a ballast, or pendulum, inside its body. Although the *Squiggle Ball* has no control system nor radio link to a ground station, it can change its direction of motion upon encountering obstacles. Even when it faces an obstacle, the ballast inside still climbs up, thanks to the rubber band employed outside the ball, preventing slippage between the ball and the obstacle. When the ballast reaches the uppermost position, the ball rolls backwards, changing the direction of motion due to the slight extension of the rubber band outwards from the body. In this manner, the *Squiggle Ball* can change its direction of motion automatically and even make its way out from a dead end in a maze.

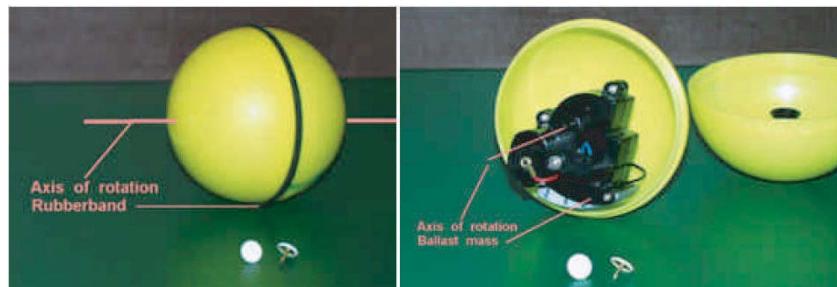


Figure 2.5: *The Squiggleball* (reproduced from Ylikorpi, 2005)

The *Mars Ball*

Another example is the *Mars Ball* proposed by Ylikorpi (2005) at the Helsinki University of Technology. It has a *two-degree-of-freedom* pendulum connected to the midpoint of the main shaft, which also shares the main rotation axis. The

main feature of the robot is that it utilizes the winds on Mars for its secondary propulsion. For this purpose, fins are attached to the spherical body, which is made up of several arc wires. The fins are for harnessing available wind power. The wired arcs enable a light-weight body and also offer the potential of a deployable structure. The *Mars Ball* has a large body (6m in diameter for the flight model, 1.3m for the prototype model) to overcome the rough terrain of Mars.



Figure 2.6: *The Mars Ball (Ylikorpi, 2005)*

The *IWBS* Robot

Koshiyama (1992a) has also developed a pendulum-driven type spherical robot. Unlike most of the robots with only interior sensors, such as inertial sensors and magnetic sensors, his rolling robot, called the *IWBS* Robot, *Inside Wheel Balancing System Robot*, has employed some exterior sensors to measure status precisely. However, putting exterior sensors may ruin some of the benefits of spherical robots, such as complete sealing of the electronics from the outside environment, and robustness due to the complete spherical shape. Koshiyama's robot, nevertheless, can still be used for indoor applications. (Koshiyama and Yamafuji, 1992a) (Koshiyama and Yamafuji, 1992b) (Koshiyama and Yamafuji, 1992c)

The *IWBS* robot just looks like a head with a set of headphones on it. The headphones part is an arch used for controlling the motion of the robot. In addition, this arch part can be stabilized and even be used for carrying humans or other items, which is normally not possible for other spherical robots, because they do not have any unrotating parts accessible from outside.

The two rods sticking out from the *ears* of the robot are called the *Contacting Rods*. The tip of the rods are always touching and sweeping the floor. In this way, measurements can be performed with respect to the floor. Each rod has two sensors. One is a rotary encoder measuring the travel distance of the robot. The other one is an angular sensor measuring the inclination of the rod and yielding the roll angle of the ball itself.

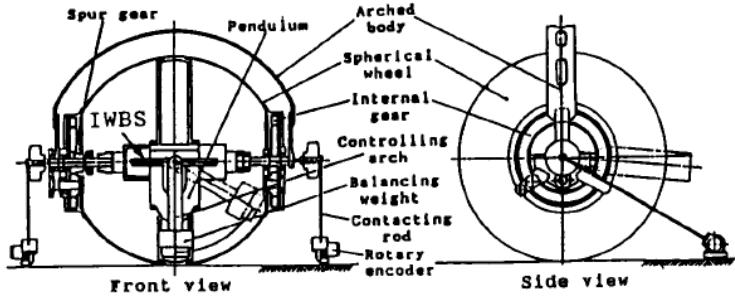


Figure 2.7: The IWBS Robot (reproduced from Koshiyama 1992a)

These measurements are used for closed-loop control of the robot. Koshiyama's paper shows that the stabilization and smooth motion control was successful using those measurements.

Although the concept and design of pendulum-driven robots is suitable for planetary exploration, the previous studies lack mathematical dynamics modeling and its evaluation using hardware. These are described later in greater detail in this study.

2.2.3 Deformable Body

Another locomotion system employs something known as a deformable or inflatable body.

The *tumbleweed ball*

One example is the *tumbleweed ball* developed by the JPL (Jet Propulsion Laboratory) in the US. (Jones, 2001) It is a large inflatable ball, similar to a giant beach ball, with scientific instruments inside its 6m-diameter body. Instead of

using a moving mass inside, it is purely propelled by the winds on Mars. Very simple motion control is achieved by simply adjusting the inflatable body: it could be stopped by partial deflation and moved again with full inflation.

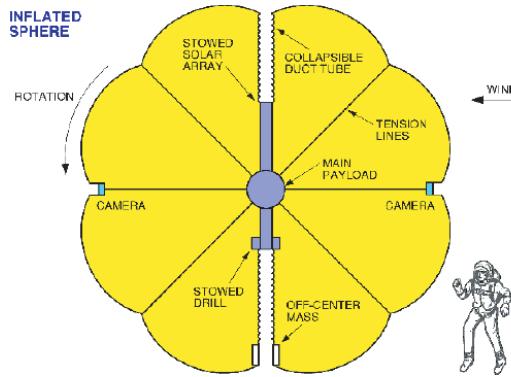


Figure 2.8: *The "Tumbleweed Ball"* (reproduced from Jones 2001)

Koharo Robot

Another interesting deformable robot is *Koharo* developed at the Ritsumeikan University in Japan. (Sugiyama et al., 2005) *Koharo* is capable of locomotion on rough terrain by rolling and bouncing. It has a deformable soft body and flexible actuators made of shape memory alloy, and can roll and bounce on the ground simply by changing the shape of the soft body.

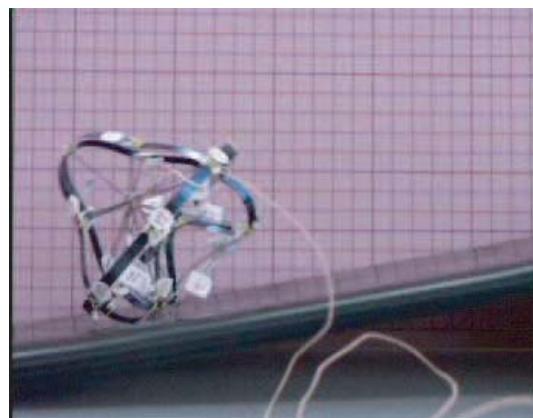


Figure 2.9: *Soft body spherical robot Koharo* (copied from the Koharo project website at Ritsumeikan University, Japan, <http://www.ritsumei.ac.jp/se/hirai/research/softrobot-j.html>)

Furthermore, the *Koharo* robots were proudly exhibited at the 2005 Aichi World

Exposition in Japan. At the exposition, a number of cutting edge technologies were presented, including robotics, automobiles, and environmental engineering. The author himself has visited Aichi and witnessed first hand the workings of the "Koharo" robots.

2.2.4 *Gyrover*

The *Gyrover* is a disk-shaped rolling robot developed at the Carnegie Mellon University that utilizes the gyroscopic effect of a spinning mass inside for stability. (Tsai et al., 1999) Although the *Gyrover* is not strictly a spherical robot, studying the *Gyrover* is useful to deepen the understanding of the motion of spherical robots.



Figure 2.10: *Gyrover* (copied from the *Gyrover* project website, <http://www.cs.cmu.edu/cyberscout/gyrover.html>)

The main feature of the *Gyrover* is its spinning mass, or gyroscope, inside the body. The basic locomotion system for moving forwards and backwards is similar to that of a pendulum-driven spherical robot, where a pendulum is used for driving. The steering principle, however, is different from either IDU-driven or pendulum-driven robots. The *Gyrover* uses a servo motor attached to the spinning gyroscope that tilts the rotation axis of the gyroscope, resulting in a steering motion of the body. The fundamental equation for these dynamics is the gyroscopic precession equation:

$$\tau = J\omega \times \Omega \quad (2.1)$$

where τ is the torque on the gyroscope, J is the moment of inertia of the gyroscope, ω is the angular rate of the gyroscope, and Ω is the precession rate. Based on this precession equation, the turning motion of the robot can be generated by applying moment on the gyroscope, as shown in Figure 2.11.

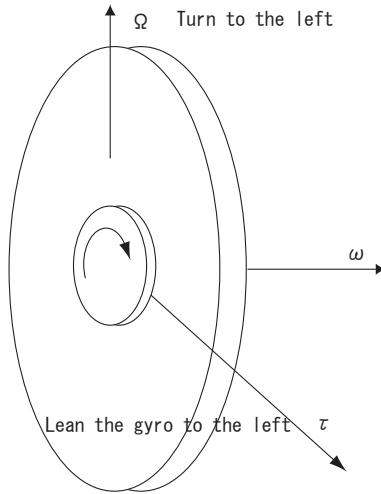


Figure 2.11: *Basic Principle for Turning*

2.3 Dynamics Analysis

As with other areas of earlier research, it is necessary to review the dynamics analysis in order to have full background knowledge of the processes involved.

2.3.1 Analysis of an IDU-driven Robot

With Newton's Law of Motion

Halme (1996) has conducted a basic motion analysis for an IDU-driven spherical robot. (Halme and Wang, 1996) First, the structure of the robot has been simplified as is shown on the left side of Figure 2.12. Here each part is represented by a point mass and moment of inertia around its center of gravity (CG). (On the right side of Figure 2.12)

Based on this model, the motion analysis has been conducted using the following motion equation for rotation:

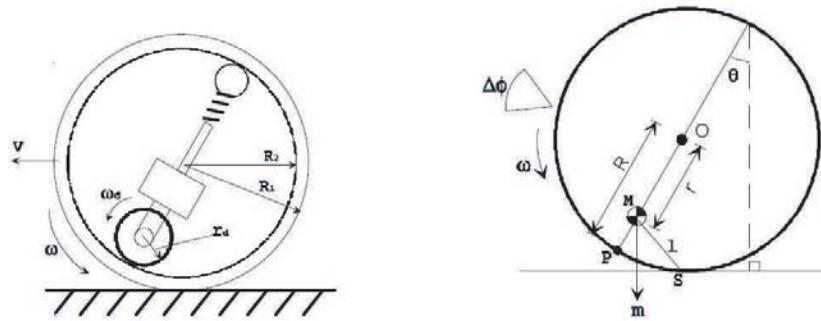


Figure 2.12: Modeling (reproduced from Halme, 2005a)

$$T = J \frac{d^2\phi}{dt^2} \quad (2.2)$$

where, T is torque, J is moment of inertia, and ϕ is rotation angle. The equation above can be expressed as follows:

$$rmg \sin \theta = I_{ball}\alpha \quad (2.3)$$

where, r is the distance between the CG of the ball and the CG of the IDU, m is the mass of the IDU, θ is the tilting angle, or the inclination of the IDU, I_{ball} is the moment of inertia of the ball excluding the IDU around the contact point on the ground (note: not the moment of inertia around the center of gravity of the ball), g is the acceleration of the gravity, and α is the rotational acceleration, or the 2nd derivative of the rotational angle of the ball. This motion analysis was conducted for the following cases:

- Driving on an even plane
- Driving uphill
- Driving over an obstacle
- Driving taking into account rolling friction

This analysis was only for driving, or moving forward. Motion analysis for steering was not performed.

With Lagrange Equation

Koshiyama (1996) has also conducted a basic motion analysis for an IDU driven robot. He separated the driving and steering motions for simplicity. He used Lagrange equations to yield motion equations for both driving and steering cases. (Koshiyama and Fujii, 1996)

Since the driving motion analysis is almost the same as the analysis in Section 3.1, this section only demonstrates the dynamics analysis for the turning motion of an IDU-driven robot.

The motion equations for turning can be yielded in the following manner.

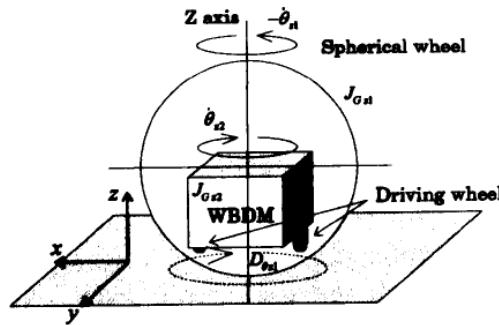


Figure 2.13: *Steering Motion Modeling (reproduced from Koshiyama, 1996)*

Energies (potential, kinetic, and rotational) for both the ball and the IDU are given as follows:

$$U_1 = U_2 = K_1 = K_2 = 0 \quad (2.4)$$

$$T_1 = \frac{1}{2}J_{z1}\omega_{z1}^2, \quad T_2 = \frac{1}{2}J_{z2}(\omega_{z1} + \omega_{z2})^2 \quad (2.5)$$

U_1 : Potential energy of the ball with respect to the height of its centroid

U_2 : Potential energy of the pendulum with respect to the height of the ball's centroid

K_1 : Kinetic energy of the ball

K_2 : Kinetic energy of the pendulum

T_1 : Rotational energy of the ball

T_2 : Rotational energy of the pendulum

J_{z1} : Moment of inertia of the ball around its vertical axis

J_{z2} : Moment of inertia of the IDU around its vertical axis

θ_{z1} : Rotation angle of the ball around its vertical axis

θ_{z2} : Rotation angle of the IDU with respect to the ball around its vertical axis

ω_{z1} : Angular rate of the ball around its vertical axis

ω_{z2} : Angular rate of the IDU with respect to the ball around its vertical axis

The Lagrange equation is given by:

$$\begin{aligned} L &= K_1 + K_2 + T_1 + T_2 - U_1 - U_2 \\ &= \frac{1}{2} J_{z1} \omega_{z1}^2 + \frac{1}{2} J_{z2} (\omega_{z1} + \omega_{z2})^2 \end{aligned} \quad (2.6)$$

hence, the Lagrange equation of motion can be calculated to be:

$$\begin{aligned} T_z - T_{zf} &= \frac{d}{dt} \left(\frac{\partial L}{\partial \omega_{z1}} \right) - \frac{\partial L}{\partial \theta_{z1}} = \frac{d}{dt} \left(\frac{\partial L}{\partial \omega_{z1}} \right) \\ &= a_{z2} J_{z2} + a_{z1} (J_{z1} + J_{z2}) \end{aligned} \quad (2.7)$$

$$\begin{aligned} -T_z &= \frac{d}{dt} \left(\frac{\partial L}{\partial \omega_{z2}} \right) - \frac{\partial L}{\partial \theta_{z2}} = \frac{d}{dt} \left(\frac{\partial L}{\partial \omega_{z2}} \right) \\ &= a_{z2} J_{z2} + a_{z1} + J_{z2} \end{aligned} \quad (2.8)$$

where T_z is torque applied between the ball and the IDU, T_{zf} is torque coming from the friction between the ball and the ground, a_{z1} is the angular acceleration of the robot around its vertical axis, and a_{z2} is the angular acceleration of the IDU around its vertical axis.

Koshiyama also conducted an experiment of steering-while-driving which determined the relationship between the steepness of steering and the amount of inner-mass displacement of a spherical robot. This is purely experimental without any theoretical dynamics investigation to support the author's claims. (Koshiyama and Yamafuji, 1992c) This experiment is very important, because, unlike an IDU-driven robot, the motion of a pendulum-driven spherical robot cannot be categorized separately into driving and steering. When it steers, both driving and steering occur simultaneously like the motion of a normal car.

2.3.2 Analysis of pendulum driven robot

Laplante (2004) offered a mathematical motion analysis of steering-while-driving for pendulum driven robots. Since the analysis is almost the same as the one in Section 3.2.3, this section gives only a short comment on the analysis.

One unclear issue of this study is that it assumes that the turning radius depends on the roll angle of the robot, without taking into account any gyro effect of the rotating body. This assumption is correct when the gyro effect can be disregarded. If a gyro effect exists, which tends to hold the direction of the rotating shaft, the trajectory of the robot may not follow the estimated trajectory of this study. Moreover, the study conducted by Laplante includes no practical hardware experiments. (Laplante, 2004)

2.4 Applications

2.4.1 Playing Toy

Michaud (2000) has developed a spherical toy robot, which is implicitly appealing for children because they are used to playing with ball-shaped objects, such as rubber, plastic, or inflatable balls. In addition, the round shape has nothing sticking out, which may hurt children or be broken easily. This robot is called the *Roboll*, and employs a pendulum-propelling system. (Michaud and Caron, 2000)

The motion of the *Roboll* is behavior based, offering four types of motion, "emergency", "spin", "straight", and "cruise", which can be activated or deactivated according to events. These simplified motions would be enough for toys.

Experiments with children have also been conducted. The tests showed that children became interested in the *Roboll* and started to play with it. The playing with the children also revealed that the robustness of the robot is significantly important because children often throw toys on the floor.

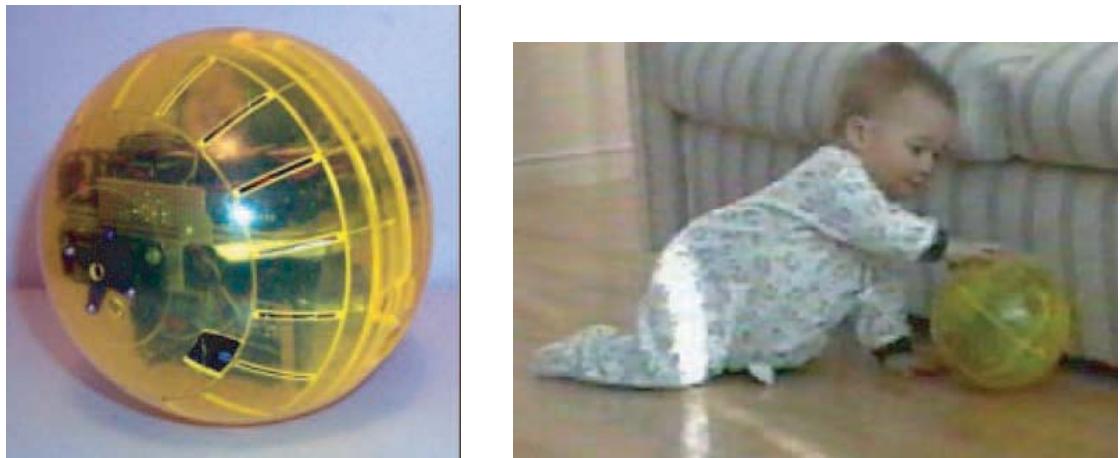


Figure 2.14: *Roball*(left), *A boy playing with a Roball*(right)(both reproduced from Michaud 2000)

2.4.2 Surveillance

Another application for spherical robots is surveillance. Rotundus, a Swedish company founded in 2004, has developed the pendulum-driven surveillance spherical robot shown in Figure 2.15. (Knight, 2005) Surveillance robots have to be able to navigate all types of terrain, such as mud and snow. Spherical robots offer great advantages for those terrain types. The robot has a radar as well as some other sensors for detecting intruders, and an alarm for reporting.



Figure 2.15: *Surveillance Robot* (reproduced from Knight, 2005)

2.4.3 Reconnaissance

Cyclops is a miniature spherical mobile robot designed for remote surveillance and reconnaissance in an urban area. (Chemel et al., 1999) The motivation for the spherical miniature design came from the great demand for police or military missions in urban areas. The small size allows covert surveillance in an enclosed area. In addition, remote surveillance enables people to survey a dangerous situation from afar. It is also sufficiently small in dimensions and light in weight to be carried by hand. The locomotion and control system of the *Cyclops* robot is mainly based on an on-board sensing system listed below:



Figure 2.16: *Cyclops* (reproduced from Chemel et al., 1999)

The locomotion system has two degrees of freedom, having two motors both along its vertical and horizontal axes. One motor along the vertical axis is for turning the heading of the robot. When the motor rotates a mass embedded in the robot, the spherical body itself turns in the opposite direction due to the reaction torque. The other motor along the horizontal axis is for moving forwards and backwards. By rotating the same mass to either the front or the back in the spherical body, the robot can rotate forwards or backwards thanks to the torque generated by the gravity.

The control and power system consists of micro-controllers, a miniature transceiver, and a power system. The micro-controllers control the peripheral components such as gyro, compass, and power system. The transceiver links the robot to the ground station operated by the human. The power system including batteries is completely self-contained in the robot's spherical body, allowing it to move freely.

The sensing system in this context means sensors for surveillance, excluding any sensors for control and locomotion. The main sensor for *Cyclops* is a subminiature black-and-white camera. The vision of the camera is transmitted to the ground station, enabling the operator to control the robot.

The robot is still, however, a prototype and there are future improvements that can be made. For instance, a further miniaturized high-resolution omni-directional camera is currently being developed. A high power video transmitter is also under development. In addition, the improvement of the whole robot in size, power, and robustness is now being planned.



Figure 2.17: *Conical Mass* (reproduced from Chemel et al., 1999)

2.4.4 Space Exploration

The Massachusetts Institute of Technology (MIT) in the US is developing spherical robots that can move by bouncing and rolling. (Young, 2006) These robots can be sent to other planets arranged altogether in an egg-carton-like container.(See figure 2.18)

The advantage of using dozens of micro robots is that they can cooperate to achieve their collective goal. For example, given correct positioning, they can relay messages back to the central unit even from deep within a cave.

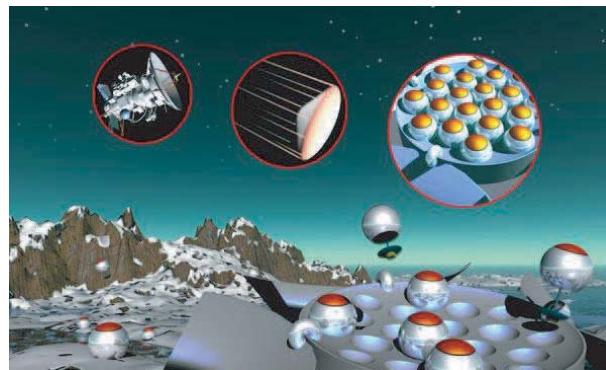


Figure 2.18: *Spherical Micro Robots sent to Mars* (reproduced from Young, 2006)

2.4.5 Entertainment

Already in 2002, the Sony Corporation has developed a spherical robot for entertainment purposes called *Q.taro* — which is short for "Quasi-stable Traveling and Action Robot". (Nakamura, 2002)



Figure 2.19: *Entertainment Robot Q.taro* (reproduced from Nakamura 2002)

Q.taro has an impressive 36 sensors in total. It switches itself on and off by sensing human hands, and avoids obstacles automatically. On top of all that, in scenarios utilizing numerous *Q.taro* robots, they start to communicate and cooperate with each other. They also recognize voice and noise.

Chapter 3

Theory

3.1 Dynamics of Driving

First of all, the modeling and dynamics analysis of the driving motion is explained. The Lagrange equation is utilized to derive the motion equations. The analysis of Koshiyama in 1996 has been modified and corrected for pendulum-driven spherical robots.

3.1.1 Horizontal Motion

Energies (potential, kinetic, and rotational) for both the ball and the pendulum of a robot are given as follows:

$$\begin{aligned} U_1 &= 0, \quad U_2 = -M_2 g e \cos(\theta_1 + \theta_2), \quad K_1 = \frac{1}{2} M_1 (r\omega_1)^2 \\ K_2 &= \frac{1}{2} M_2 \left\{ (r\omega_1 - e \cos(\theta_1 + \theta_2)(\omega_1 + \omega_2))^2 + (e \sin(\theta_1 + \theta_2)(\omega_1 + \omega_2))^2 \right\} \\ T_1 &= \frac{1}{2} J_1 \omega_1^2, \quad T_2 = \frac{1}{2} J_2 (\omega_1 + \omega_2)^2 \end{aligned} \tag{3.1}$$

U_1 : Potential energy of the ball with respect to the height of its centroid

U_2 : Potential energy of the pendulum with respect to the height of the ball's centroid

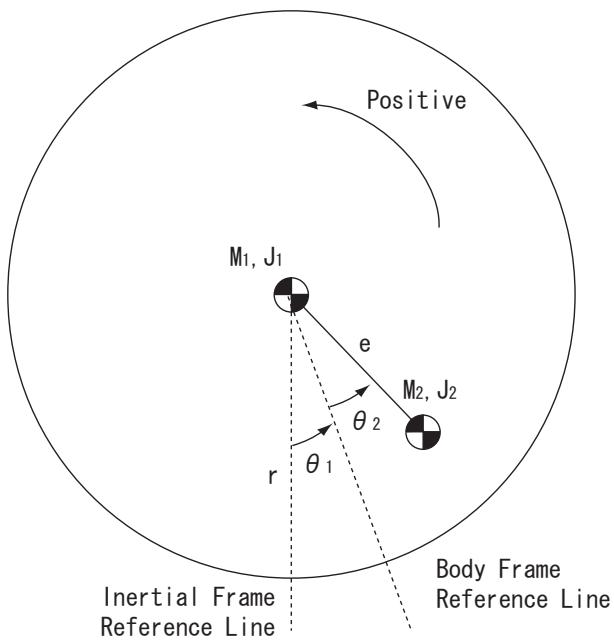


Figure 3.1: Modeling of the ball (driving)

K_1 : Kinetic energy of the ball

K_2 : Kinetic energy of the pendulum

T_1 : Rotational energy of the ball

T_2 : Rotational energy of the pendulum

r : Radius of the ball

e : Distance between centroids of the ball and the pendulum

θ_1 : Rotation angle of the ball

θ_2 : Rotation angle of the pendulum with respect to the ball

ω_1 : Angular rate of the ball

ω_2 : Angular rate of the pendulum with respect to the ball

J_1 : Moment of inertia of the ball around its centroid

J_2 : Moment of inertia of the pendulum around its centroid

M_1 : Mass of the ball

M_2 : Mass of the pendulum

g : Acceleration of gravity

Here, the Lagrange function can be calculated as follows:

$$L = K_1 + K_2 + T_1 + T_2 - U_1 - U_2 \quad (3.2)$$

Now, the Lagrange equation of motion can be expressed as follows:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \omega_1} \right) - \frac{\partial L}{\partial \theta_1} = -T + T_f \quad (3.3)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \omega_2} \right) - \frac{\partial L}{\partial \theta_2} = T \quad (3.4)$$

where t is time, T is torque applied between the ball and the pendulum, and T_f is the torque coming from the friction between the ball and the ground. Therefore, we have:

$$\begin{aligned} -T + T_f &= a_1(J_1 + J_2 + M_1r^2 + M_2r^2 + M_2e^2 - 2M_2re \cos(\theta_1 + \theta_2)) \\ &\quad + a_2(J_2 - M_2re \cos(\theta_1 + \theta_2) + M_2e^2) \\ &\quad + M_2re \sin(\theta_1 + \theta_2)(\omega_1 + \omega_2)^2 + M_2ge \sin(\theta_1 + \theta_2) \end{aligned} \quad (3.5)$$

$$\begin{aligned} T &= a_1(J_2 - M_2re \cos(\theta_1 + \theta_2) + M_2e^2) + a_2(J_2 + M_2e^2) \\ &\quad + M_2ge \sin(\theta_1 + \theta_2) \end{aligned} \quad (3.6)$$

For simplicity, the tilting angle of the pendulum with respect to the vertical line, namely $\theta_2 - \theta_1$, is called the *driving angle* in this paper.

3.1.2 Uphill and Downhill Motion

Once the horizontal motion equations have been calculated, it is easy to derive dynamics for uphill and downhill motion. If it is taken that the only difference between the horizontal motion and the uphill or downhill motion is the direction of the gravity, there is a need for a slight change in the values of U_1 and U_2 as mentioned in the section above.

$$U_1 = -M_1gr\theta_1 \sin \alpha \quad (3.7)$$

$$U_2 = -M_2gr\theta_1 \sin \alpha - M_2ge \cos(\theta_1 + \theta_2 + \alpha) \quad (3.8)$$

where α is the inclination of the slope.

$$\begin{aligned}
 -T + T_f &= a_1(J_1 + J_2 + M_1r^2 + M_2r^2 + M_2e^2 - 2M_2re \cos(\theta_1 + \theta_2)) \\
 &\quad + a_2(J_2 - M_2re \cos(\theta_1 + \theta_2) + M_2e^2) \\
 &\quad + M_2re \sin(\theta_1 + \theta_2)(\omega_1 + \omega_2)^2 \\
 &\quad + M_2ge \sin(\theta_1 + \theta_2 + \alpha) - (M_1 + M_2)gr \sin \alpha
 \end{aligned} \tag{3.9}$$

$$\begin{aligned}
 T &= a_1(J_2 - M_2re \cos(\theta_1 + \theta_2) + M_2e^2) + a_2(J_2 + M_2e^2) \\
 &\quad + M_2ge \sin(\theta_1 + \theta_2 + \alpha)
 \end{aligned} \tag{3.10}$$

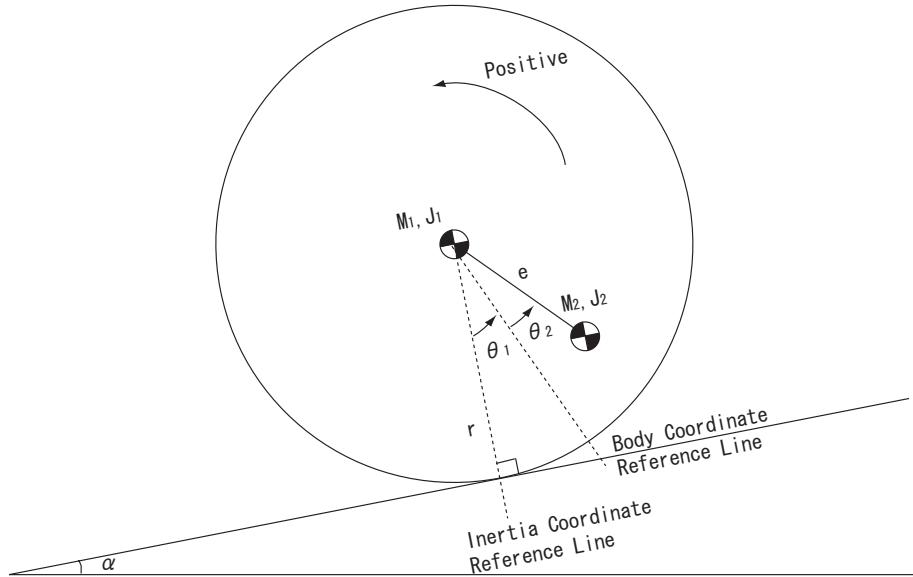
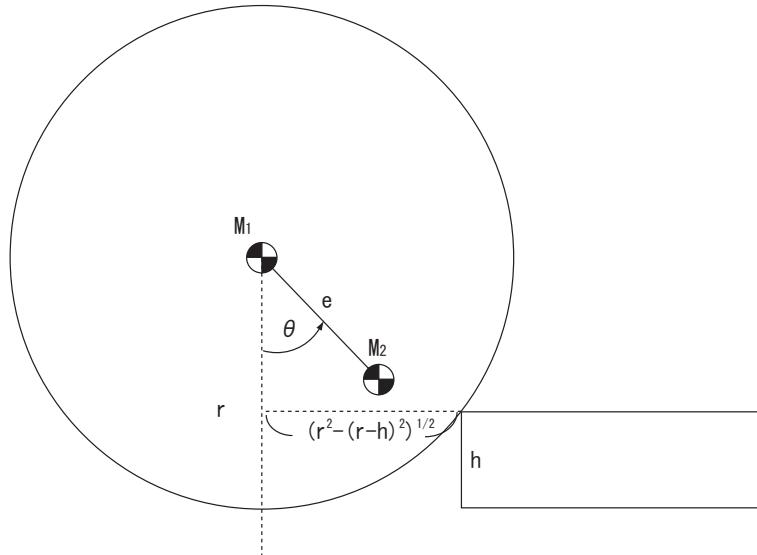


Figure 3.2: Modeling of the ball (uphill/downhill)

3.1.3 Obstacle crossing

This section shows the obstacle crossing motion of the robot. Figure 3.3 depicts the static configuration of a spherical robot encountering an obstacle. It is assumed here that no slippage and no momentum are taken into account.

In order to climb up the obstacle, the driving torque must be greater than the

Figure 3.3: *Obstacle crossing*

counter torque by the gravity as shown in the following equations.

$$\begin{aligned}
 & (\text{Driving torque}) > (\text{Counter torque}) \\
 & M_2 g e \sin \theta > (M_1 + M_2) g \sqrt{r^2 - (r - h)^2} \\
 \therefore & M_2^2 e^2 \sin^2 \theta > (M_1 + M_2)^2 \{r^2 - (r - h)^2\} \\
 \therefore & (r - h)^2 > r^2 - \frac{e^2 \sin^2 \theta M_2^2}{(M_1 + M_2)^2} \\
 \therefore & h < r - \sqrt{r^2 - \left(\frac{e \sin \theta M_2}{M_1 + M_2}\right)^2} \\
 \therefore & h_{max} = r - \sqrt{r^2 - \left(\frac{e M_2}{M_1 + M_2}\right)^2} \quad (3.11)
 \end{aligned}$$

where h is the height of the obstacle, θ is the driving angle of the pendulum. By substituting the parameters for the prototype robot developed in this study, the maximum height of an obstacle which can be driven over by the robot can be calculated to be $0.0012\text{m} = 1.2\text{mm}$.

Compared to the robot radius of 22cm, 1.2mm is significantly small, meaning obstacle crossing is a critical weakness in the spherical robot motion. Therefore, some solutions for obstacle crossing would be needed. These include taking advantage of the momentum of the robot, using a "jumping" by moving the pendulum rapidly, and utilizing a wired body of which the "opening" allows to cross over an obstacle even without touching it.

3.2 Dynamics of Steering

3.2.1 Modeling

Figure 3.4 shows the modeling of the steering motion.

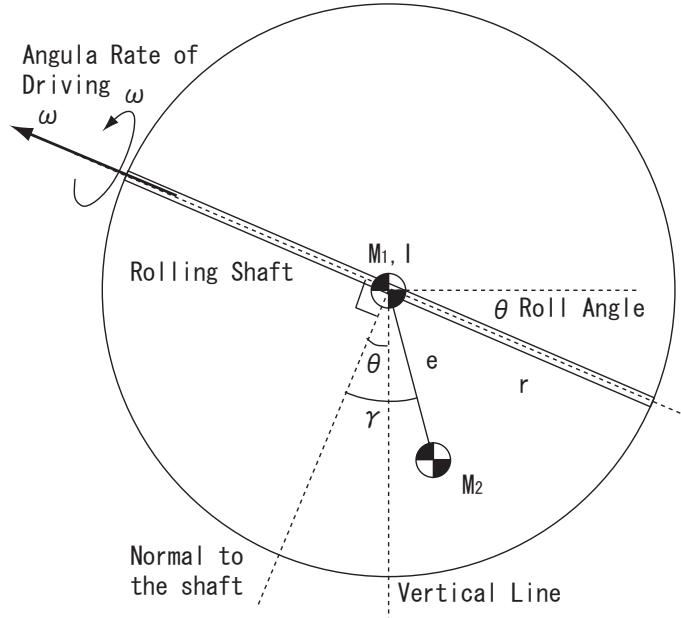


Figure 3.4: Modeling for steering

θ : Roll angle of the ball

γ : Tilting angle of the pendulum with respect to the ball

ω : Angular rate of driving of the ball

Ω : Angular rate of steering

r : Radius of the ball

e : Distance between the centroids of the ball and the pendulum

I : Moment of inertia of the ball around its roll axis

M_1 : Mass of the ball

M_2 : Mass of the pendulum

g : Acceleration of the gravity

For simplicity, the sideways tilting angle of the pendulum with respect to the ball, namely γ , is called the *steering angle* in this paper. Note that the inclination of the main shaft, or θ , is called the *roll angle*.

3.2.2 Ball traveling at low speed

The basic principles for this analysis as well as the analysis in Section 3.2.3 are:

- Equilibrium of force and torque including the centrifugal force of steering
- The robot follows the circumference where the center point is the intersection of the floor and the line of the rotating shaft of the robot.

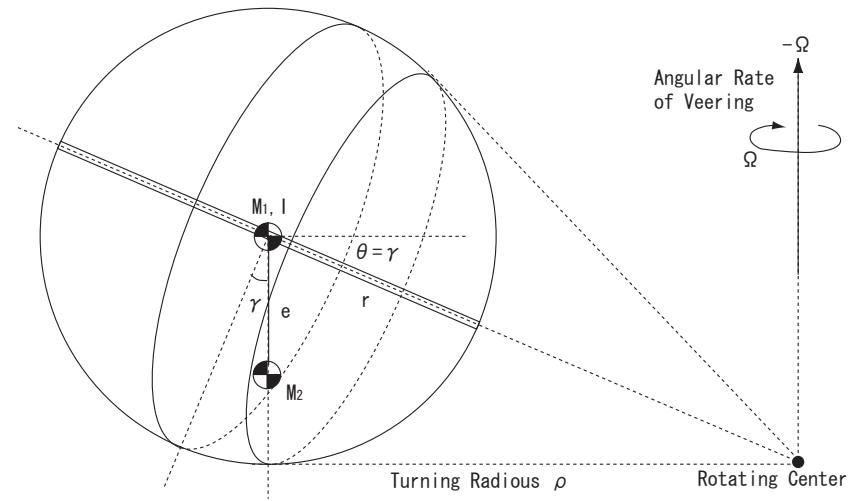


Figure 3.5: Modeling for steering with low speed

Centrifugal force can be disregarded in the calculation process when the robot travels at very low speed. In such cases, the roll angle θ is the same as the steering angle of the pendulum γ . The steering radius ρ can be calculated as:

$$\rho = \frac{r}{\tan \gamma} \quad (3.12)$$

The angular rate for steering Ω is:

$$(Path Length) = \omega r \cos \gamma = \rho \Omega \quad (3.13)$$

$$\begin{aligned} \Omega &= \frac{\omega r \cos \gamma}{\rho} \\ &= \frac{\omega r \cos \gamma \tan \gamma}{r} \\ &= \omega \sin \gamma \end{aligned} \quad (3.14)$$

3.2.3 Ball traveling at intermediate speed

When the robot travels at intermediate speed, the gyro effect can be disregarded but the centrifugal force, however, has still to be taken into account.

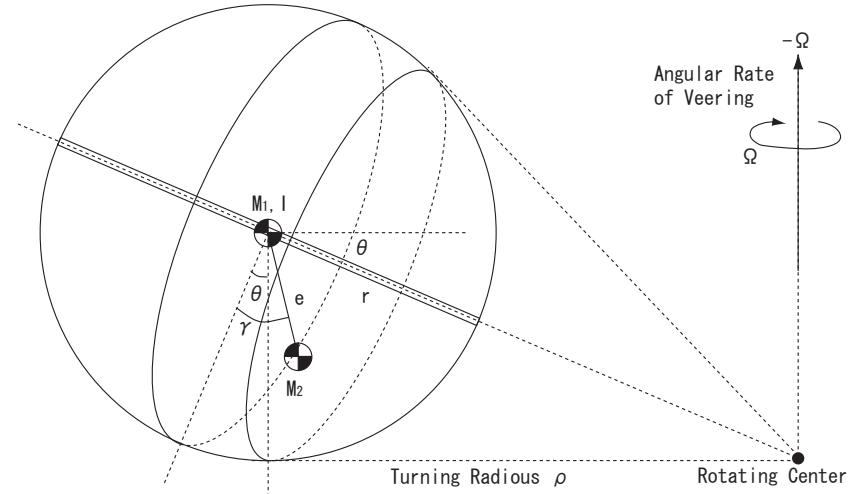


Figure 3.6: Modeling for steering with intermediate speed

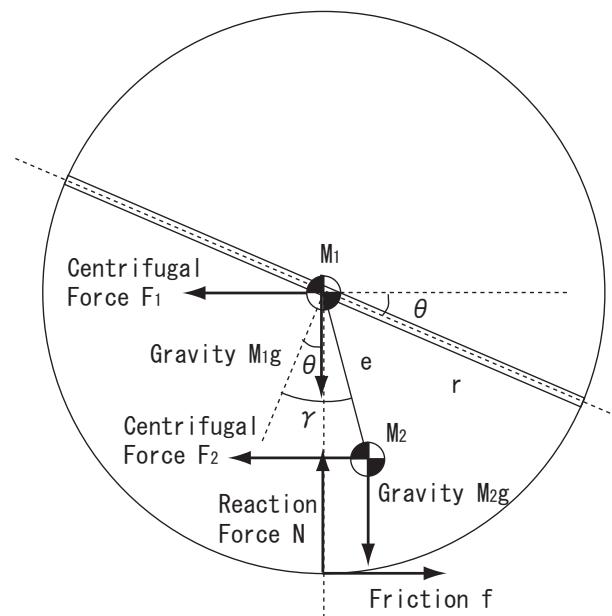


Figure 3.7: Forces on each element

F_1 : Centrifugal force on the centroid of the ball

F_2 : Centrifugal force on the centroid of the Pendulum

N : Vertical reaction force from the contact point

f : Friction at the contact point

$$F_1 = M_1 \rho \Omega^2 \quad (3.15)$$

$$F_2 = M_2 (\rho - \sin(\gamma - \theta)e) \Omega^2 \quad (3.16)$$

$$\Omega = \frac{v}{\rho} \quad (3.17)$$

Equilibrium of force and torque:

$$F_1 + F_2 = f \quad (3.18)$$

$$N = (M_1 + M_2)g \quad (3.19)$$

$$rf = e \sin(\gamma - \theta) M_2 g + e \cos(\gamma - \theta) F_2 \quad (3.20)$$

From the round shape of the sphere:

$$\tan \theta = \frac{r}{\rho} \quad (3.21)$$

For simplicity, it can be assumed that θ and $\gamma - \theta$ are sufficiently small thus allowing approximation as follows:

$$\sin(\gamma - \theta) \approx (\gamma - \theta) \quad (3.22)$$

$$\cos(\gamma - \theta) \approx 1 \quad (3.23)$$

$$\tan \theta \approx \theta \quad (3.24)$$

In addition, by disregarding the multiplication of small values, the following equation is obtained:

$$\rho = \frac{(M_1 + M_2)rv^2 + eM_2(gr - v^2)}{egM_2\gamma} \quad (3.25)$$

As you might have noticed, the error involved in the simplification (3.22), (3.23), and (3.24) is not always small. For example, according to the test results shown

in Figure 7.5, $\tan \theta$ can be as much as 0.3, which leads to an error of 3% in simplification (3.24). The error would be up to 10% depending on the turning, making Equation (3.25) unreliable to some extent.

Nevertheless, the control methods in this study does not rely on Equation (3.25) at all. The most important characteristic of a spherical robot is that the more the steering angle is and the less the speed is, the steeper the turning becomes. Using this characteristic, the robot can control itself by feedback control.

Therefore, the error of several percentage will not destroy the control methods and the main results of this study.

3.2.4 Ball traveling at high speed

When the ball travels at high driving speed, neither the gyro effect nor the centrifugal force can be neglected. Here, $\rho \gg r > e$ is approximated. Frictional force at the contact point is:

$$\begin{aligned} f &= F_1 + F_2 \\ &= M_1\rho\Omega^2 + M_2(\rho - \sin(\gamma - \theta)e)\Omega^2 \\ &\approx (M_1 + M_2)\rho\Omega^2 \end{aligned} \quad (3.26)$$

Torque applied to the ball is:

$$\begin{aligned} N &= e \sin(\gamma - \theta)M_2g + e \cos(\gamma - \theta)F_2 - rf \\ &\approx M_2ge \sin(\gamma - \theta) + M_2\rho\Omega^2e \cos(\gamma - \theta) - r(M_1 + M_2)\rho\Omega^2 \end{aligned} \quad (3.27)$$

$$\begin{aligned} \boldsymbol{\omega} &= \Omega\hat{z} - \dot{\theta}\hat{l} + \omega\hat{k} \\ &= \Omega(\cos\theta\hat{m} - \sin\theta\hat{k}) - \dot{\theta}\hat{l} + \omega\hat{k} \quad (\because \hat{z} = \cos\theta\hat{m} - \sin\theta\hat{k}) \\ &= \Omega\cos\theta\hat{m} + (\omega - \Omega\sin\theta)\hat{k} \quad (\because \dot{\theta} = 0) \end{aligned} \quad (3.28)$$

$$\therefore \boldsymbol{L} = I\Omega\cos\theta\hat{m} + I_3(\omega - \Omega\sin\theta)\hat{k} \quad (3.29)$$

$$\begin{aligned} \therefore \boldsymbol{N} &= -N\hat{l} = \frac{d\boldsymbol{L}}{dt} = \boldsymbol{\Omega} \times \boldsymbol{L} \\ &= I\Omega^2\cos\theta\hat{z} \times \hat{m} + I_3\Omega(\omega - \Omega\sin\theta)\hat{z} \times \hat{k} \\ &= I\Omega^2\cos\theta(-\sin\theta)\hat{l} + I_3\Omega(\omega - \Omega\sin\theta)(-\cos\theta)\hat{l} \\ &= -\hat{l}(I\Omega^2\cos\theta\sin\theta + I_3\Omega\cos\theta(\omega - \Omega\sin\theta)) \end{aligned} \quad (3.30)$$

$$\begin{aligned}
\therefore N &= I\Omega^2 \cos \theta \sin \theta + I_3 \Omega \cos \theta (\omega - \Omega \sin \theta) \\
&= I_3 \Omega \omega \cos \theta \left(1 + \frac{I - I_3}{I_3} \frac{\Omega}{\omega} \sin \theta\right) \\
&= I_3 \Omega \omega \cos \theta \quad (\because \frac{I - I_3}{I_3} \ll 1, \frac{\Omega}{\omega} \ll 1)
\end{aligned} \tag{3.31}$$

ω : Angular velocity vector of the ball

Ω : Angular rate of the revolution of the ball

θ : Roll angle of the ball

L : Angular momentum vector of the ball

I_3 : Moment of inertia of the ball around its spinning axis

I : Moment of inertia of the ball around the axis perpendicular to the spinning axis

N : Torque vector applied to the ball

Equating (3.27) and (3.31), we get:

$$M_2 g e \sin(\gamma - \theta) + M_2 \rho \Omega^2 e \cos(\gamma - \theta) - r(M_1 + M_2) \rho \Omega^2 \approx I_3 \Omega \omega \cos \theta \tag{3.32}$$

There is also another equation on the path length:

$$\begin{aligned}
\Omega &= r \omega \cos \theta \\
\therefore \Omega &= \frac{r \omega \cos \theta}{\rho} \\
I_3 \frac{r \omega \cos \theta}{\rho} \omega \cos \theta &\approx M_2 g e \sin(\gamma - \theta) + M_2 \rho \frac{r^2 \omega^2 \cos^2 \theta}{\rho^2} e \cos(\gamma - \theta) \\
&\quad - r(M_1 + M_2) \rho \frac{r^2 \omega^2 \cos^2 \theta}{\rho^2} \\
\therefore I_3 \frac{r \omega^2 \cos^2 \theta}{\rho} &\approx M_2 g e \sin(\gamma - \theta) + M_2 \frac{r^2 \omega^2 \cos^2 \theta}{\rho} e \cos(\gamma - \theta) \\
&\quad - r(M_1 + M_2) \frac{r^2 \omega^2 \cos^2 \theta}{\rho} \\
\therefore \rho M_2 g e \sin(\gamma - \theta) &\approx r \omega^2 \cos^2 \theta (I_3 - M_2 r e \cos(\gamma - \theta) + r^2 (M_1 + M_2)) \\
\therefore \rho &\approx \frac{r \omega^2 \cos^2 \theta (I_3 - M_2 r e \cos(\gamma - \theta) + r^2 (M_1 + M_2))}{M_2 g e \sin(\gamma - \theta)}
\end{aligned}$$

If $\theta = 0$ is stabilized by the gyro effect, the following equation is obtained:

$$\rho \approx \frac{r \omega^2 (I_3 - M_2 r e \cos \gamma + r^2 (M_1 + M_2))}{M_2 g e \sin \gamma} \tag{3.33}$$

3.2.5 Numerical Calculation

Using the following parameters for the prototype robot, it is possible to calculate the steering radius for several different conditions. Note that the zero speed in Figure 3.8 means infinitively slow speed. It can also be interpreted as the steering characteristics disregarding centrifugal force completely.

$$\begin{aligned} e &= 0.065m, \quad g = 9.8m/s^2, \quad M_1 = 3.294kg, \quad M_2 = 1.795kg \\ r &= 0.226m, \quad I_3 = J_1 = 6.33 \times 10^{-2}kgm^2, \quad v = r\omega \end{aligned}$$

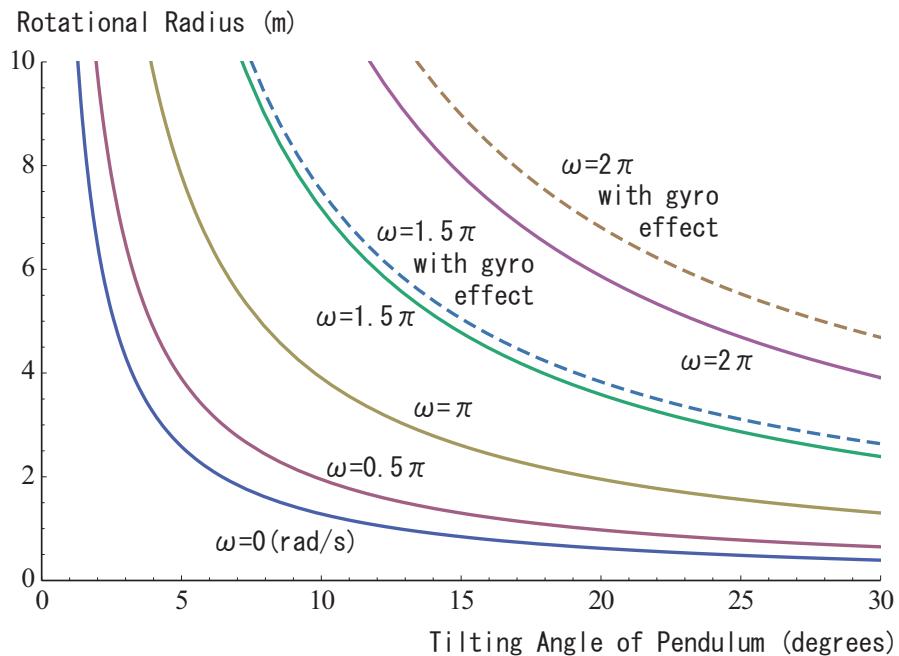


Figure 3.8: *Rotational radius against pendulum steering angle*

As shown in Figure 3.8, the steering is steeper with larger steering angles and slower speeds.

The most important fact revealed here is that with or without the gyro effect, the basic trend does not change at all; the slower the speed and the greater the steering angle of the pendulum, the steeper the steering would be. Since this trend remains the same with or without the gyro effect, the basic strategy for steering can be consistent.

Chapter 4

Prototype Robot

4.1 System Overview

The objective of the development of a prototype robot is to investigate the applied control methods. The system consists of the following two main components: the prototype robot and the ground station (GS). The prototype robot is equipped with an on-board computer (OBC), a power source, a communication system, and some sensors of which the data is used for feedback control. The prototype also has motors for locomotion.



Figure 4.1: *Prototype robot*

The ground station is for processing and displaying the sensor data, and for controlling higher level functioning, such as "set speed", "brake", "turn left" or "turn right". For controlling lower level functioning such as "accelerate to the commanded speed" or "keep the orientation of the robot", the OBC uses the control methods proposed in this study. The following graph depicts the system layout of the whole system.

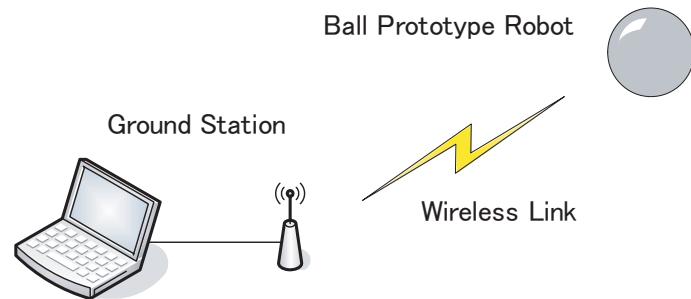


Figure 4.2: System Overview

4.2 Electronic Architecture

The electronic structure comprising the ground station (GS) can be represented as shown in the following block diagram.

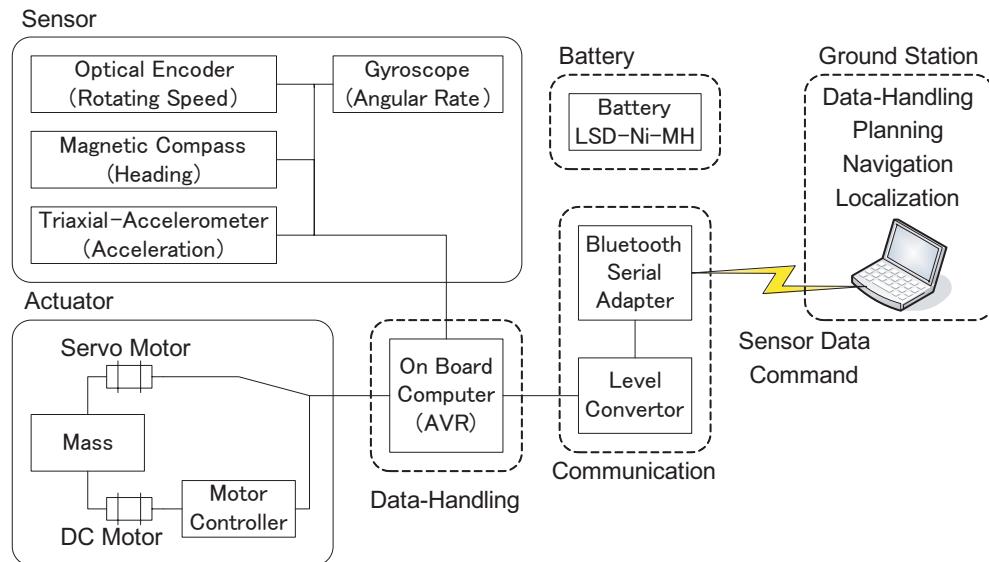


Figure 4.3: Block Diagram

4.2.1 Block Diagram Description

On-Board Computer and Level Converter

The on-board computer is the Crumb128-CAN module. It is a module that combines an Atmel AT90CAN128 AVR 8-bit microcontroller with a standard serial port with an RS232 transceiver and a USB2.0 device interface. The main components used to provide the necessary features used in this project are as follows:

- 8-channel 10-bit analog-to-digital converter (ADC) for acquiring data from analogue-output sensors
- Dual Programmable Serial Universal Synchronous-Asynchronous Receiver/Transmitter (USARTs) for communication with the ground station via a wireless serial link
- 128kbytes of on-chip, non-volatile flash memory for program code storage

Wireless System

The robot is equipped with the IOGEAR Class1 Bluetooth Serial Adapter GBA301 for wireless communication with the ground station. It allows Bluetooth wireless connection of RS232 serial equipment up to a range of 100 meters. The transfer rate selected is 9600 baud, the slowest one, as it provides the best quality.

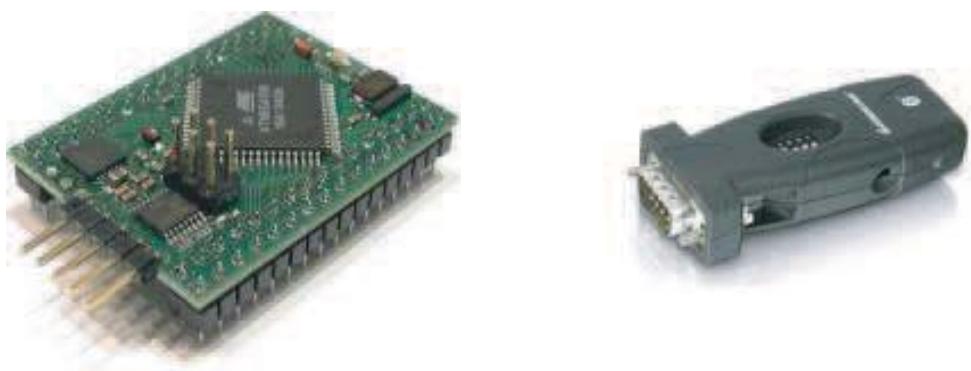


Figure 4.4: *The Crumb128-CAN Module (left), The IOGEAR Bluetooth Serial Adapter (right)*

4.2.2 Magnetic Compass

A magnetic compass is implemented to determine the direction of travel of the robot. The component is the Hitachi HM55B Compass Module with a dual-axis magnetic field sensor. It also makes all the power and signal connections accessible in a breadboard-friendly 6-pin dual inline package (DIP). The significant features are:

- Sensitivity to microtesla variations in magnetic field strength
- Simplified direction by resolving magnetic field measurements into two-component axes
- Capability for 6-bit (64-direction) resolution measurements after software calibration
- Relatively short measurement time (30 to 40 ms)

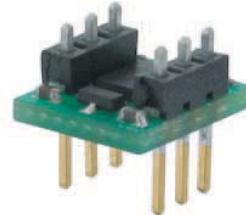


Figure 4.5: *The Hitachi HM55B Compass Module*

In order to avoid the electromagnetic noise from the main electronic board and motors, the compass is attached at a distance from them at the tip of a plastic rod. (See Figure 4.6)

4.2.3 Tri-axial Accelerometer

The Kionix tri-axial accelerometer KXM52-1050 is mounted to measure the acceleration of the pendulum. The main features of the device are as follows:

- Package: 8-pin DIP
- Output: Analogue 660mV/G
- Offset Voltage: 1.65V
- Supply Voltage: 3.3 – 5V

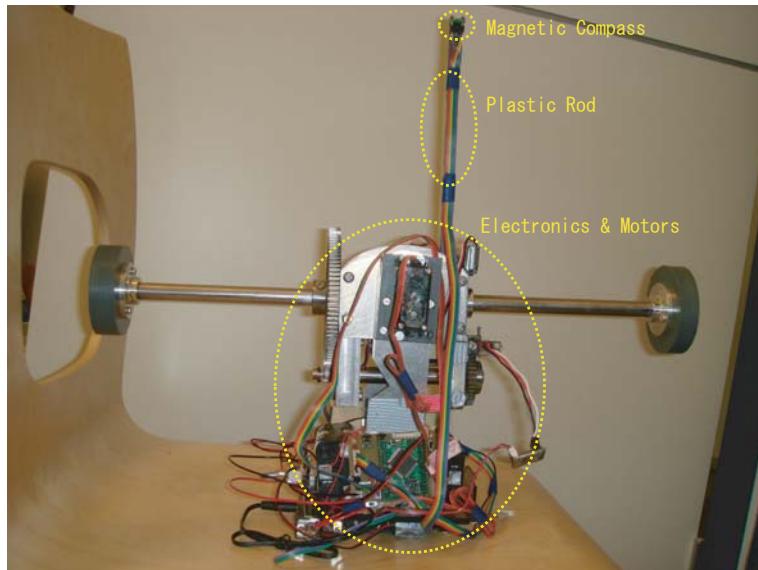


Figure 4.6: *The position of the compass*

4.2.4 Gyroscope

The Kondo KRG-3 Gyro is used to detect the angular velocity of the roll angle of the ball, or the main shaft. It is extremely small in size, light-weight, and of single axis detection. The main features of the device are as follows:

- Angular Rate Range: $\pm 300\text{deg/sec}$
- Output: Analogue $0.67\text{mV}/(\text{degree/sec})$
- Offset Voltage: 1.35V
- Supply Voltage: $3 - 5\text{V}$
- Weight: 1.8g
- Size: $22.5 \times 12.5 \times 7.8 \text{ mm}$

4.2.5 Servo Motor

Two servo motors are used to tilt the pendulum inside the spherical robot. The bodies of the servo motors are mounted on the pendulum, and the arms are fixed to the main shaft of the robot. In this way, the rotational motion of the motors tilts the direction of the pendulum. The servos used in the tests are the



Figure 4.7: The Kionix Tri-axial Accelerometer KXM52-1050 (left), The Kondo KRG-3 Gyro (right)

BlueBird BMS660MG-HS high-torque metal-gear servos made in Taiwan, which were originally designed for racing toy cars. The main features of the device are as follows:

- Angular Rate Range: $\pm 300\text{deg/sec}$
- Torque at 4.8V : 13kg cm
- Ball Bearing: Dual
- Supply Voltage: 4.8 – 6.0V
- Weight: 54g
- Size: 40.5 x 20.0 x 42.5 mm

4.2.6 DC Motor

A DC motor is used to continuously rotate the pendulum mass for driving the robot. The Mabuchi RS540SH DC motor is used. The main features of the device are as follows:

- Nominal Voltage: 12V
- Speed (No Load): 17500 rpm
- Current (No Load): 0.95A
- Weight: 160g
- Size: $35.8 \phi \times 50.0$ mm

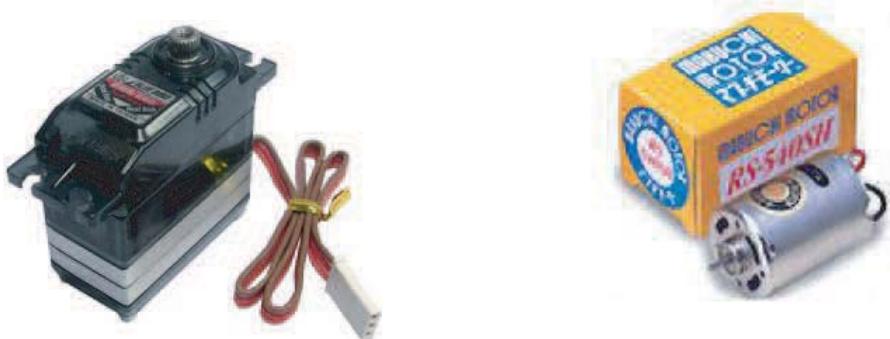


Figure 4.8: *The BlueBird BMS660MG-HS servo motor (made in Taiwan)(left), The Mabuchi RS540SH DC Motor (right)*

4.2.7 Battery

Low self-discharge NiMH batteries (LSD NiMH) are used as the power supply. The LSD NiMH battery is the newest technology introduced in 2005. The batteries have significantly reduced self-discharge, thereby being always ready-to-use after charging. They also have a reduced memory effect, allowing charging them even after slight discharge, making management of the charging process easy. These features are suitable for testing the robots intermittently and charging batteries whenever convenient.

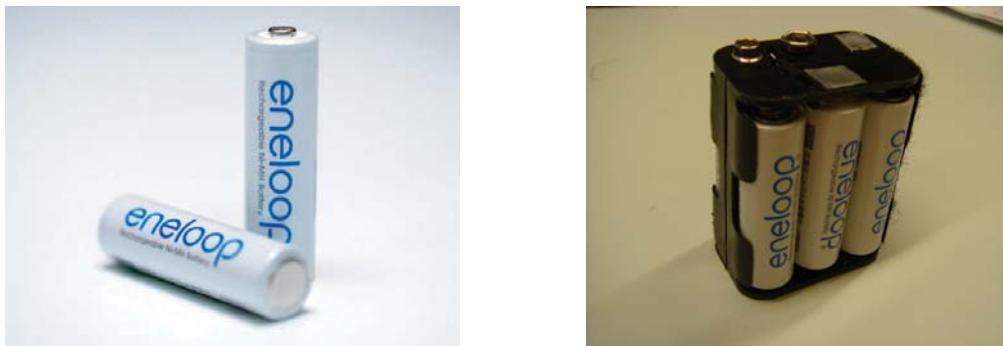


Figure 4.9: *Sanyo Low self-discharge NiMH battery Eneloop (left), 6-AA-battery pack used for the prototype (right)*

4.2.8 Motor Controller

In order to control speed and torque of the DC motor by adjusting the supply voltage, a motor controller is used, connected between the main board and the

DC motor. This motor controller is a standard controller designed and developed by the Automation and Systems Department, TKK. The controller receives the following inputs:

- Enable: To enable the control (0V or 5V)
- Direction: To set the direction of rotation (0V or 5V)
- (Pulse Width Modulation) PWM: To control the speed or torque by adjusting the voltage (0V to 5V, maximum frequency 30kHz)

4.2.9 Optical encoder

The optical encoder mounted on the prototype is a handmade one using a photo coupler, which was not in use in the lab where testing was carried out. Unfortunately, the manufacturer and the product name for the photo coupler are unknown. The photo coupler and its peripheral circuit gives 0V when the light path of the coupler is blocked, and otherwise 3.4V. The rotating disk has eight blades, intermittently blocking the light path while rotating, generating eight pulses per rotation of the motor. The AVR microchip constantly monitors the pulses and increments a certain parameter every rising and falling edge of the pulse, thus incrementing 16 ticks per rotation of the motor.

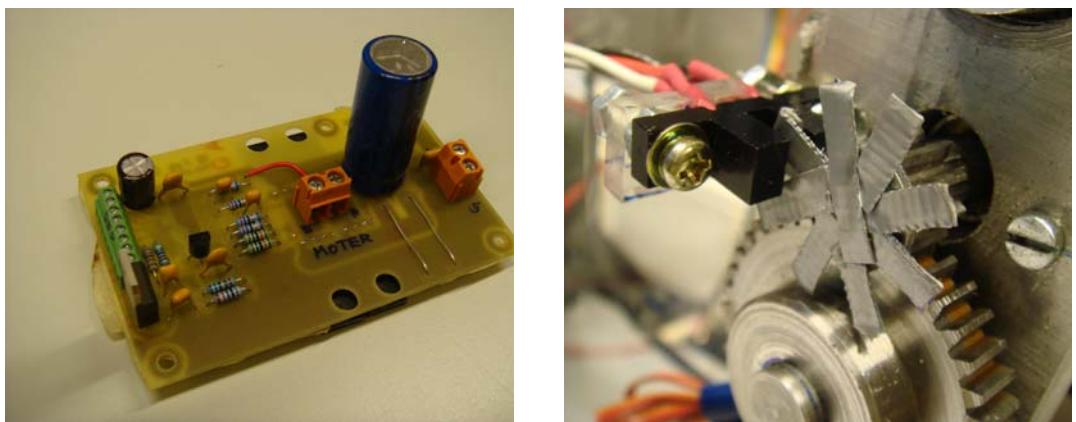


Figure 4.10: *The standard motor controller used at TKK (left), The handmade optical encoder with a photo coupler (right)*

4.2.10 Main board

The main electronic board is a hub connecting all the sensors, power supplies, communication adapter and motors. It also has some LEDs indicating the on/off status of the main board, servo motors, and the AVR microchip itself.

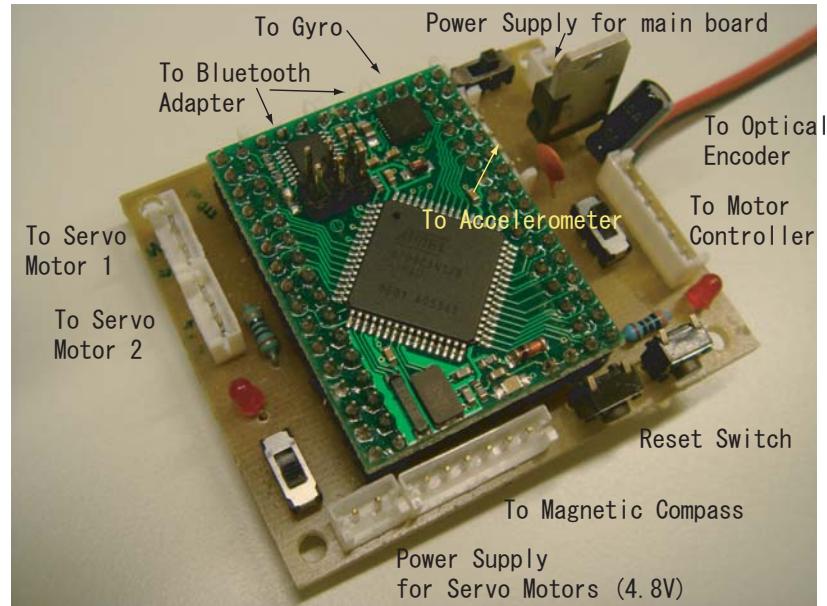


Figure 4.11: *The main board*

4.2.11 Main switch

The main switch receives 7.2V from two 6-AA-battery boxes and 4.8V from one 4-AA-battery box, and supplies 7.2V to the motor controller, 7.2V to the main board in which a regulator converts it to 5.0V, and 4.8V to the servo motors with only one switch.

4.2.12 Mounting

A plastic case is attached at the bottom of the aluminum pendulum, holding battery boxes, the main board, the motor controller, and the Bluetooth adapter altogether.

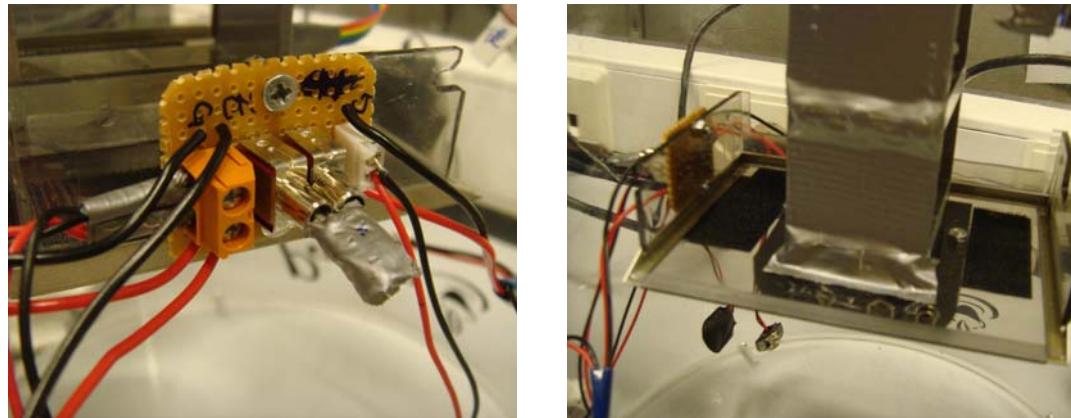


Figure 4.12: *The main switch (left), The holder of the electronics (right)*

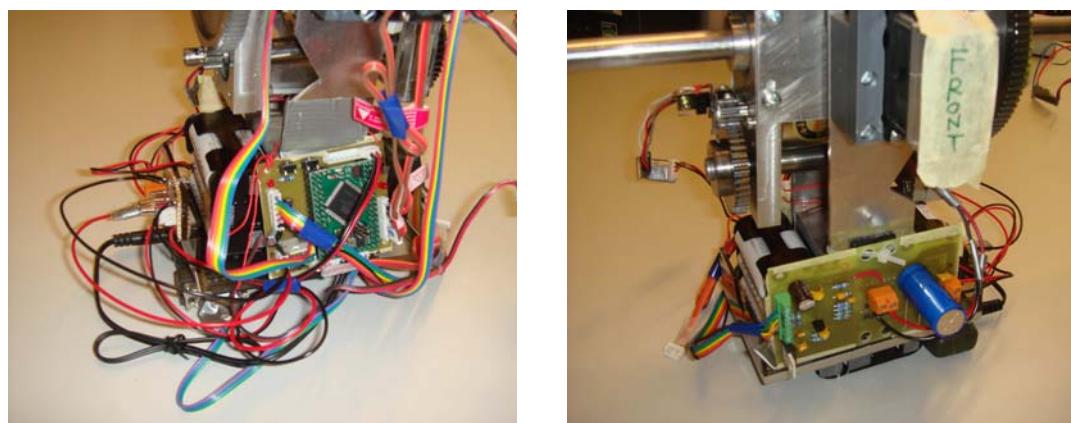


Figure 4.13: *The whole electronic system(left: front view, right: rear view)*

4.3 Specifications

This section describes the specifications of the prototype robot and components.

Table 4.1: Specifications of the pendulum

Parameters	Values	Symbols
Total mass	1795g	M_2
Moment of inertia (around rotation axis)	$0.007428kgm^2$	J_2
Moment of inertia (around horizontal line normal to shaft)	$0.007728kgm^2$	J_{2s}
Distance between ball center and pendulum CG	0.065m	e

Table 4.2: Specifications of the ball

Parameters	Values	Symbols
Mass of right hemisphere	985g	
Mass of left hemisphere with plastic stripe	1230g	
Left plastic (connecting main shaft and ball)	96g	
Right plastic (connecting main shaft and ball)	106g	
Total mass (excluding pendulum)	3294g	M_1
Moment of inertia (around rotation axis)	0.0633kgm^2	J_1
Moment of inertia (around horizontal line normal to shaft)	0.0777kgm^2	J_{1s}
Radius	0.226m	r

Table 4.3: Specifications of the DC motor

Parameters	Values	Symbols
Mass	158g	
No load speed	17500rpm	
Stall torque	230mNm	
Torque constant	0.00852 Nm/A	K_t
Speed constant	0.00638 Vs/rad	K_e
Terminal resistance	0.324 Ω	R

Table 4.5: Specifications of the electronics

Table 4.4: Specifications of the battery

Parameters	Values
Mass of battery box (6xAA)	168g
Mass of battery box (4xAA)	114g
Typical capacity	2000mAh
Minimum capacity	1900mAh
Voltage	1.2V

Parameters	Values
Mass of Crumb128AVR	32g
Mass of motor driver	52g
Mass of RS232 connector	6g
Mass of Bluetooth adapter	22g
Mass of the servo motor	54g

Chapter 5

Control Methods: Driving

The use of the term "control methods" in this study is taken to refer to control *supporting* methods for manual control. First, this section describes control supporting methods for the driving motion of the prototype robot.

5.1 Background

Manual control of the spherical robot is difficult due to the spherical shape of the robot. The robot displays clumsy motion when no control supporting methods are implemented. Therefore, some kind of control supporting system is necessary. In order to achieve this, however, there are two difficulties that must be taken into account. The first one is that the conditions of the terrain to be driven over is not always predictable. For instance, the terrain might be uphill, downhill, rocky, or flat. This means that the dynamics modeling is not always useful for controlling the robot. Especially, the friction T_f cannot be determined if the condition of the terrain is not fixed. Therefore, control supporting methods should be effective even on different types of terrain.

Another difficulty is that sensors cannot be mounted on the sphere in order to measure its status directly. All electronics and sensors must be mounted onto or wired to the pendulum of the robot, thus making it impossible to mount sensors on the outer spherical surface of the robot. Therefore, the status of the pendulum

with respect to the sphere (using an optical encoder) and to the outer world (using a accelerometer, compass, and gyro) can be measured, but not the status of the sphere with respect to the ground it travels over.

As a result, feedback control using the status of the ball with respect to the ground is not possible. For example, it is not possible to measure the parameter θ_1 , which is the rotated angle of the sphere with respect to the ground. Instead, only the θ_2 can be measured, which is the rotated angle of the pendulum with respect to the ball.

Consequently, some control methods using only the status inside the ball, namely the rotated angle of the pendulum with respect to the ball, and without fixed dynamic modeling are necessary.

5.2 Stability of the robot rotation

This study proposes the following three control methods.

1. Control by supply voltage to the motor
2. Control by motor rotation speed
3. Control by torque of the motor

In fact, the first one, controlling by supply voltage, needs no control supporting system at all. The voltage to the DC motor is the lowest level input to the robot without any processing. Alternatively, the other two control methods need some kind of control supporting system. This system has to somehow realize the targeted rotation speed or torque by adjusting the applied voltage to the DC motor.

Assuming voltage, speed, and torque are completely controllable, let us consider if the control of the ball can be stable or not. The transfer functions are derived from the input parameters (voltage, speed, and torque) to the rotation of the ball, θ_1 . These functions allows evaluation whether the system is stable or not. As there are no sensors on the outer surface of the ball, the open-loop transfer functions are derived here.

5.2.1 Voltage to Ball Rotation

In order to analyze the relationship between the supply voltage and the rotation of the ball, the motor should be modeled.

DC motor modeling

A DC motor can be modeled with two basic equations; One is the equation of voltage, and the other is the equation of torque and current.

$$V = IR + K_e \omega \quad (5.1)$$

$$T = K_t I \quad (5.2)$$

I: Current flowing through the armature

R: Resistance of the motor

V: Supplied voltage

K_e: Speed constant

T: Generated torque

ω: Rotating velocity

K_t: Torque constant

By eliminating the current *I* from the equations, it is possible to arrive at the following equation:

$$\begin{aligned} V &= \frac{T}{K_t} R + K_e \omega_2 \\ \therefore T &= \frac{K_t}{R} (V - K_e \omega_2) \end{aligned} \quad (5.3)$$

Friction modeling

The friction in this section means the one between the robot and the surface driven over by the robot. The friction of the ball rotation, T_f , consists of two major types of friction. One is the Coulomb friction, which is almost constant in spite of the change in rotation speed. The other is the viscous friction, which is proportional to the speed of the ball. Here we model T_f as:

$$T_f = T_c + T_v \omega_1 \quad (5.4)$$

where T_c is the Coulomb friction, which can be both positive and negative depending on the direction of the rotation, and T_v is the coefficient for the viscous friction, which is a negative value.

Linearization of the dynamics equation

In order to derive a transfer function, the dynamics equations (3.5) and (3.6) should be simplified. First, it is assumed that the pendulum is oriented almost downwards, therefore $\theta_1 + \theta_2 \ll 1$, $\omega_1 + \omega_2 \ll 1$, and $a_1 + a_2 \ll 1$. Here you can get the linearized equations (5.5) and (5.6):

$$\begin{aligned} -T + T_f &= a_1(J_1 + J_2 + M_1 r^2 + M_2 r^2 + M_2 e^2 - 2M_2 r e) \\ &\quad + a_2(J_2 - M_2 r e + M_2 e^2) + M_2 g e (\theta_1 + \theta_2) \end{aligned} \quad (5.5)$$

$$\begin{aligned} T &= a_1(J_2 - M_2 r e + M_2 e^2) + a_2(J_2 + M_2 e^2) \\ &\quad + M_2 g e (\theta_1 + \theta_2) \end{aligned} \quad (5.6)$$

Let the Laplace transform of θ_1 and θ_2 be $\Theta_1(s)$ and $\Theta_2(s)$ respectively, those of ω_1 and ω_2 be $s\Theta_1(s)$ and $s\Theta_2(s)$, and that of T be $T(s)$, and that of V be $V(s)$. Eliminating T_f using (5.4), and disregarding T_c since it has no effect on the transfer functions in question, we can transform the equations (5.5), (5.6), and (5.3) using Laplace transform:

$$\begin{aligned} -T &= [-sT_v + egM_2 + s^2 \{ J_1 + J_2 + r^2M_1 + (e-r)^2M_2 \}] \Theta_1 \\ &\quad + [s^2J_2 + e \{ g + (e-r)s^2 \} M_2] \Theta_2 \end{aligned} \quad (5.7)$$

$$T = s^2J_2(\Theta_1 + \Theta_2) + eM_2 [\{g + (e-r)s^2\} \Theta_1 + (g + es^2)\Theta_2] \quad (5.8)$$

$$T = \frac{K_t}{R}(V - K_e s \Theta_2) \quad (5.9)$$

The transfer function can be obtained by eliminating T and Θ_2 from equations (5.7), (5.8), and (5.9):

$$\begin{aligned} \frac{\Theta_1}{V} &= \frac{A + Bs^2}{C + Ds + Es^2 + Fs^3} \quad (5.10) \\ A &= -2egK_tM_2 \\ B &= K_t(-2J_2 + e(-2e+r)M_2) \\ C &= -egRT_vM_2 + 2egK_eK_tM_2 \\ D &= -T_vK_eK_t + egRJ_1M_2 + egr^2RM_1M_2 + egr^2RM_2^2 \\ E &= -RT_vJ_2 + 2J_2K_eK_t - e^2RT_vM_2 + K_eK_t(J_1 + r^2M_1 + (2e^2 - 3er + r^2)M_2) \\ F &= e^2RJ_1M_2 + e^2r^2RM_1M_2 + RJ_2(J_1 + r^2(M_1 + M_2)) \end{aligned}$$

The characteristic equation, which can be used to calculate the poles of the system, is the denominator of the transfer function (5.10):

$$C + Ds + Es^2 + Fs^3 = 0 \quad (5.11)$$

Using the values shown in Section 4.3, the poles of the systems can be calculated as:

$$s_1 = -0.43 \quad (5.12)$$

$$s_{23} = -0.041 \pm 9.40i \quad (5.13)$$

This shows that the system is stable.

5.2.2 Motor Speed to Ball Rotation

By eliminating T from equations (5.7) and (5.8), the following transfer function can be obtained:

$$\frac{\Theta_1}{\Theta_2} = \frac{-2s^2 J_2 + e(-2g + (-2e + r)s^2) M_2}{-sT_v + 2egM_2 + s^2(J_1 + 2J_2 + r^2M_1 + (2e^2 - 3er + r^2)M_2)} \quad (5.14)$$

For our prototype robot, it is:

$$\frac{-8.33 - 0.014s^2}{8.336 + 0.510s + s^2} \quad (5.15)$$

The characteristic function is:

$$-sT_v + 2egM_2 + s^2(J_1 + 2J_2 + r^2M_1 + (2e^2 - 3er + r^2)M_2) = 0 \quad (5.16)$$

The poles are calculated as follows:

$$s = \frac{T_v \pm \sqrt{T_v^2 - 8egM_2(J_1 + 2J_2 + r^2M_1 + (2e^2 - 3er + r^2)M_2)}}{2(J_1 + 2J_2 + r^2M_1 + (2e^2 - 3er + r^2)M_2)} \quad (5.17)$$

Since $T_v < 0$, this system is stable. For the prototype robot, the poles are:

$$s_{12} = -0.26 \pm 2.88i \quad (5.18)$$

This means that the whole system is stable if the motor speed is controllable.

5.2.3 Motor Torque to Ball Rotation

By eliminating Θ_2 from equations (5.7) and (5.8), the following transfer function can be obtained:

$$\frac{\Theta_1}{T} = \frac{-2s^2 J_2 + e(-2g + (-2e + r)s^2) M_2}{esM_2((g + es^2)(-T_v + sJ_1 + r^2sM_1) + gr^2sM_2) + s^3J_2(-T_v + sJ_1 + r^2s(M_1 + M_2))}$$

The characteristic function is:

$$esM_2((g + es^2)(-T_v + sJ_1 + r^2sM_1) + gr^2sM_2) + s^3J_2(-T_v + sJ_1 + r^2s(M_1 + M_2)) = 0 \quad (5.19)$$

The poles of this system are:

$$s_1 = -0.43 \quad (5.20)$$

$$s_{23} = -0.036 \pm 9.40i \quad (5.21)$$

$$s_4 = 0 \quad (5.22)$$

This means that the whole system is stable if the motor torque is controllable.

5.3 Control method

The previous section shows that the ball rotation can be stable even if the rotation parameters are not measured. Therefore, it is enough to focus on controlling the pendulum, or the DC motor, with respect to the ball. This section describes the speed control and torque control of the DC motor.

5.3.1 Rotation Speed Control

In order to reach the target speed of the motor by controlling the supply voltage, the PID control is used with the optical encoder measurement feedback. First, the transfer function from V to Θ_2 is determined:

$$\frac{\Theta_2}{V} = \frac{K_t (-sT_v + 2egM_2 + s^2 (J_1 + 2J_2 + r^2 M_1 + (2e^2 - 3er + r^2) M_2))}{s(A + Bs + Cs^2 + Ds^3)} \quad (5.23)$$

$$\begin{aligned} A &= -egRT_v M_2 + 2egK_e K_t M_2 \\ B &= -T_v K_e K_t + egR J_1 M_2 + egr^2 R M_1 M_2 + egr^2 R M_2^2 \\ C &= -R v i s J_2 - e^2 R T_v M_2 + K_e K_t (J_1 + 2J_2 + r^2 M_1 + (2e^2 - 3er + r^2) M_2) \\ D &= R J_1 J_2 + e^2 R J_1 M_2 + e^2 r^2 R M_1 M_2 + r^2 R J_2 (M_1 + M_2) \end{aligned}$$

For the prototype, the transfer function is:

$$\frac{\Theta_2}{V} = \frac{14.35 + 0.8786s + 1.721s^2}{38.34s + 88.31s^2 + 0.5161s^3 + s^4} \quad (5.24)$$

Therefore, the transfer function from V to the motor rotation speed ω_2 is:

$$\frac{\Omega_2}{V} = \frac{14.35 + 0.8786s + 1.721s^2}{38.34 + 88.31s^1 + 0.5161s^2 + s^3} \quad (5.25)$$

where $\Omega_2 = s\Theta_2$ is the Laplace transform of ω_2 .

According to the ultimate sensitivity method, or the Ziegler-Nichols method, the gains for the PID control can be calculated as (Kano, 2005):

$$K_P = 3.66 \quad (5.26)$$

$$K_I = 0.2 \quad (5.27)$$

$$K_D = 0.05 \quad (5.28)$$

The gains for the PI control are:

$$K_P = 2.74 \quad (5.29)$$

$$K_I = 0.33 \quad (5.30)$$

Here, the PID control is utilized as shown in Figure 5.1.

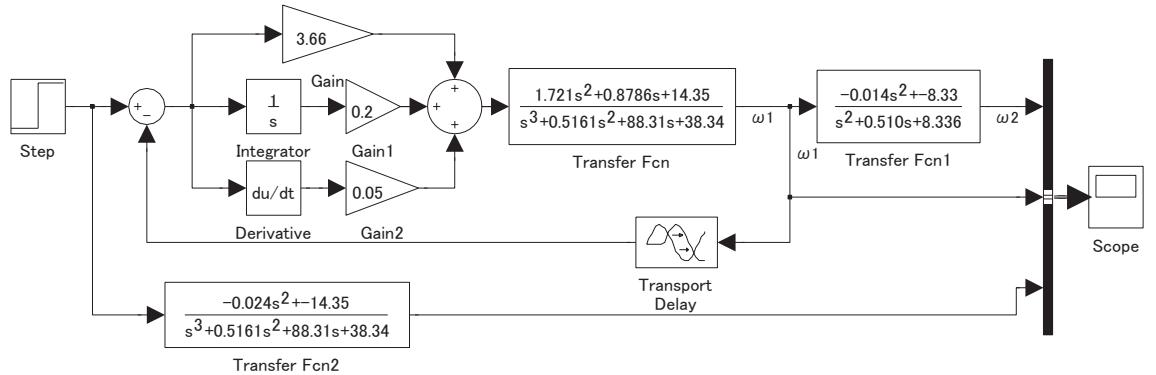


Figure 5.1: Block diagram of driving motion Stabilization

5.3.2 Torque Control

One difficulty of torque control is that the target torque cannot always be reached during operation. The applied torque should be canceled by counter torque, such as friction from outside. However, operation on smooth floor surfaces needs only small torque to maintain speed, resulting in the continuous acceleration of the robot due to the excessive torque, and the robot easily reaches its maximum speed. In doing so, this causes vibration of the driving motor, generating large noise in the accelerometer.

Therefore, torque control is not suitable for supporting manual control. For this reason, torque control has neither been designed nor implemented in the prototype.

5.3.3 Other assistant controls

Some other control methods for assisting the main control methods are also implemented in the prototype. These include: voltage increment limit, driving angle limitation, and voltage limit.

Voltage incrementation limit

In order to avoid situations where the robot accelerating too rapidly, a limit in the change rate in voltage, namely dV/dt , is implemented. This prevents the pen-

dulum from swinging around uncontrollably when it starts moving. The voltage incrementation limit is set at 3.2V/sec.

Driving angle limitation

In order to avoid the pendulum spinning around in a circle without accelerating the ball, the driving angle of the pendulum is limited. This driving angle limitation is set at 70 degrees from the vertical position of the pendulum.

Voltage limit

In order to prevent the ball from reaching extremely high speed, which may break surrounding objects and the ball itself if they collide, the supply voltage is limited to 8.0V.

Chapter 6

Control methods: Steering

There are two major issues in steering control related to the prototype spherical robot:

- How to steer
- How to stabilize the roll angle of the ball

6.1 Introduction

Even if the dynamics modeling is not completely fixed, and the status of the robot with respect to the ground is not determined, the previous model shows very important facts related to steering.

Section 3.2 shows that two input parameters can be used to control the steering rate, also known as the rotational radius: the speed of the robot and the steering angle of the pendulum. Lower speeds and larger steering angles of the pendulum lead to a larger steering rate, namely a smaller rotational radius. And vice versa, higher speeds and smaller steering angles lead to a smaller steering rate, namely a larger rotational radius.

There is, however, one difficulty in steering. The rotational radius depends also on the roll angle of the main shaft. Even though the steering angle of the pendulum

is fixed, the spherical shape of the ball causes the roll angle to oscillate, thus constantly changing the rotational radius while steering.

This situation is unfavorable for the steering control, and all oscillation should be somehow suppressed to facilitate a stable steering.

Nevertheless, when the rotation speed is high, thanks to the gyro stabilization effect, the roll angle is stable. Therefore, this section proposes a stabilization method based on modeling with a low traveling speed.

6.2 Stabilizing the roll angle

In order to achieve stable steering control during low-speed operation, the roll angle has to be stabilized. However, this stabilization control can be switched off when the ball reaches a certain rotational speed.

6.2.1 Modeling

The same motion equations (3.5), (3.6) for the driving can also be used for tilting motion, the only difference being the interpretation of each parameter.

$$\begin{aligned} T_f - T_s &= a_{1s}(J_{1s} + J_{2s} + M_1r^2 + M_2r^2 + M_2e^2 - 2M_2re \cos(\theta_{1s} + \theta_{2s})) \\ &\quad + a_{2s}(J_{2s} - M_2re \cos(\theta_{1s} + \theta_{2s}) + M_2e^2) \\ &\quad + M_2re \sin(\theta_{1s} + \theta_{2s})(\omega_{1s} + \omega_{2s})^2 + M_2ge \sin(\theta_{1s} + \theta_{2s}) \end{aligned} \quad (6.1)$$

$$\begin{aligned} T_s &= a_{1s}(J_{2s} - M_2re \cos(\theta_{1s} + \theta_{2s}) + M_2e^2) + a_{2s}(J_{2s} + M_2e^2) \\ &\quad + M_2ge \sin(\theta_{1s} + \theta_{2s}) \end{aligned} \quad (6.2)$$

r : Radius of the ball

e : Distance between centroids of the ball and the pendulum

θ_{1s} : Roll angle

θ_{2s} : Steering angle of the pendulum

ω_{1s} : Angular rate of the main shaft in tilting motion

ω_{2s} : Angular rate of the pendulum with respect to the ball in tilting motion

a_{1s} : Acceleration in tilting of the main shaft

a_{2s} : Acceleration in tilting of the pendulum with respect to the ball

J_{1s} : Moment of inertia of the ball around its centroid about the horizontal line normal to the main shaft

J_{2s} : Moment of inertia of the pendulum around its centroid about the horizontal line normal to the main shaft

M_1 : Mass of the ball

M_2 : Mass of the pendulum

g : Acceleration of gravity

T_s : Torque by servo motors

T_f : Friction between the ball and the travelling surface

The simplified equations (5.5), (5.6) can also be used:

$$\begin{aligned} -T_s + T_f &= a_{1s}(J_{1s} + J_{2s} + M_1r^2 + M_2r^2 + M_2e^2 - 2M_2re) \\ &\quad + a_{2s}(J_{2s} - M_2re + M_2e^2) + M_2ge(\theta_{1s} + \theta_{2s}) \end{aligned} \quad (6.3)$$

$$\begin{aligned} T_s &= a_{1s}(J_{2s} - M_2re \cos(\theta_{1s} + \theta_{2s}) + M_2e^2) + a_{2s}(J_{2s} + M_2e^2) \\ &\quad + M_2ge(\theta_{1s} + \theta_{2s}) \end{aligned} \quad (6.4)$$

Therefore, the same transfer function (5.14) can also be used:

$$\frac{\Theta_{1s}}{\Theta_{2s}} = \frac{-2s^2J_{2s} + e(-2g + (-2e + r)s^2)M_2}{-sT_v + 2egM_2 + s^2(J_{1s} + 2J_{2s} + r^2M_1 + (2e^2 - 3er + r^2)M_2)} \quad (6.5)$$

For the prototype robot, the transfer function can be calculated as:

$$\frac{\Theta_1}{\Theta_2} = -\frac{2.287 + 0.00443s^2}{2.287 + 0.14s + 0.289s^2} \quad (6.6)$$

The real parts of the poles of this system are negative, meaning this system is stable. However, the dampening rate of the system is very low, thus taking time for stabilization, meaning that active feedback control for quick stabilization is needed.

In order to facilitate feedback control, the roll angle has to be measured somehow. The sensors that can be used here are the accelerometer for the angle itself, and the gyro for the angular rate.

6.2.2 Determination of roll angle

Although it may have some noise, the accelerometer on board is used to determine the roll angle.

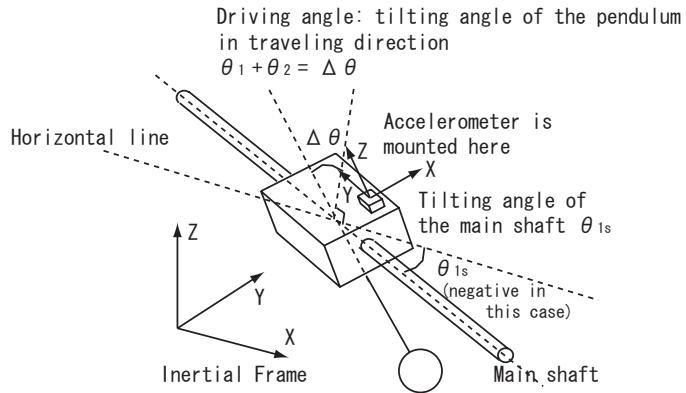


Figure 6.1: Position of accelerometer

Starting with the inertial frame, a frame rotates about the Y axis at an angle of $-\theta_{1s}$, then about the X axis at an angle of $\Delta\theta$, and, finally, about the Z axis at an angle of 90 degrees.

The final orientation of the frame, which is the accelerometer frame, is given relative to the initial frame, as:

$$\begin{aligned} {}^I_A R &= R_Y(-\theta_{1s})R_X(\Delta\theta)R_z(90^\circ) \\ &= \begin{pmatrix} \cos(\theta_{1s}) & 0 & -\sin(\theta_{1s}) \\ 0 & 1 & 0 \\ \sin(\theta_{1s}) & 0 & \cos(\theta_{1s}) \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\Delta\theta) & -\sin(\Delta\theta) \\ 0 & \sin(\Delta\theta) & \cos(\Delta\theta) \end{pmatrix} \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \quad (6.7)$$

If it is assumed that the only acceleration applied to the measurement is the acceleration of the gravity, the output of the accelerometer (g_x, g_y, g_z) can be determined as:

$$\begin{pmatrix} g_x \\ g_y \\ g_z \end{pmatrix} = {}^I_A R^T \begin{pmatrix} 0 \\ 0 \\ -g \end{pmatrix} = \begin{pmatrix} -g \cos(\theta_{1s}) \sin(\Delta\theta) \\ g \sin(\theta_{1s}) \\ -g \cos(\Delta\theta) \cos(\theta_{1s}) \end{pmatrix} \quad (6.8)$$

Now, the two parameters, θ_{1s} and $\Delta\theta$, can be estimated using the output of the accelerometer as:

$$\theta_{1s} = \arcsin\left(\frac{g_y}{g}\right) \quad (6.9)$$

$$\Delta\theta = \arctan\left(\frac{g_x}{g_z}\right) \quad (6.10)$$

6.2.3 Stabilization Method

The modified differential control method is used for the stabilization of the prototype robot. The term "modified" means that the differentiation is not applied after the subtraction of the feedback from the input, but applied to the feedback value before subtraction.

The reason why differential control is solely used here is that only a dampening effect, which is proportional to the speed, is needed to reduce the oscillation.

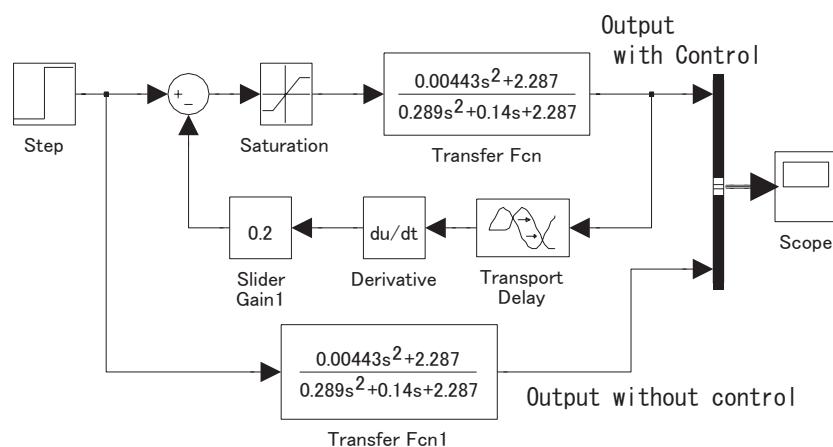


Figure 6.2: Block Diagram of Sideways Motion Stabilization

Chapter 7

Simulation and Test Results

7.1 Driving Control

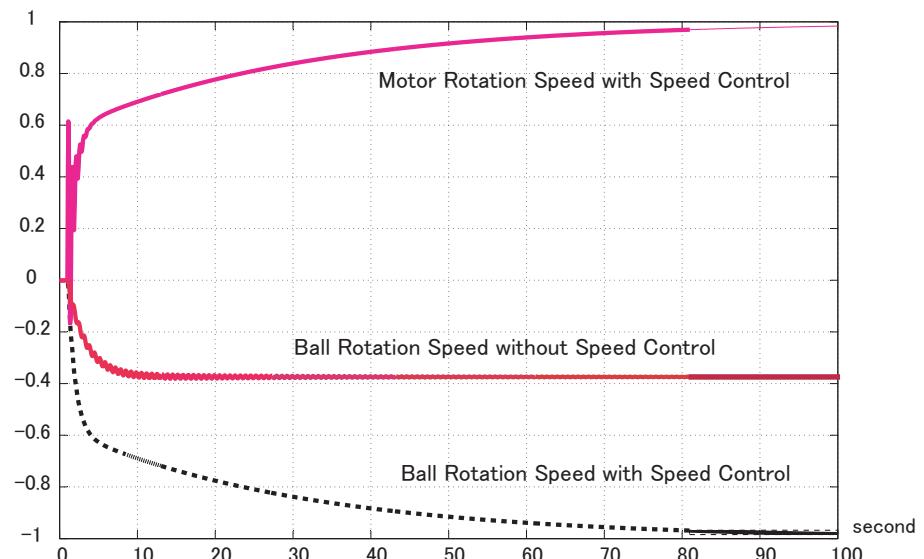


Figure 7.1: *Simulation result of the motor speed control*

Figure 7.1 depicts the simulation result of the speed control. The model and parameters of this simulation are based on the block diagram shown in Figure 5.1 with the time delay of 0.1 seconds. The input is the step function with the amplitude of 1, which is supposed to generate the motor speed of 1. Figure 7.1 shows the motor speed reaching 1 thanks to the speed control. The two lines in

the lower half of the figure denote the ball rotation speed with and without the speed control.

Even without the speed control, the rotational speed can be stabilized, which means that the speed control is not necessarily needed. Nevertheless, in order to reach the target speed, the speed control can be used.

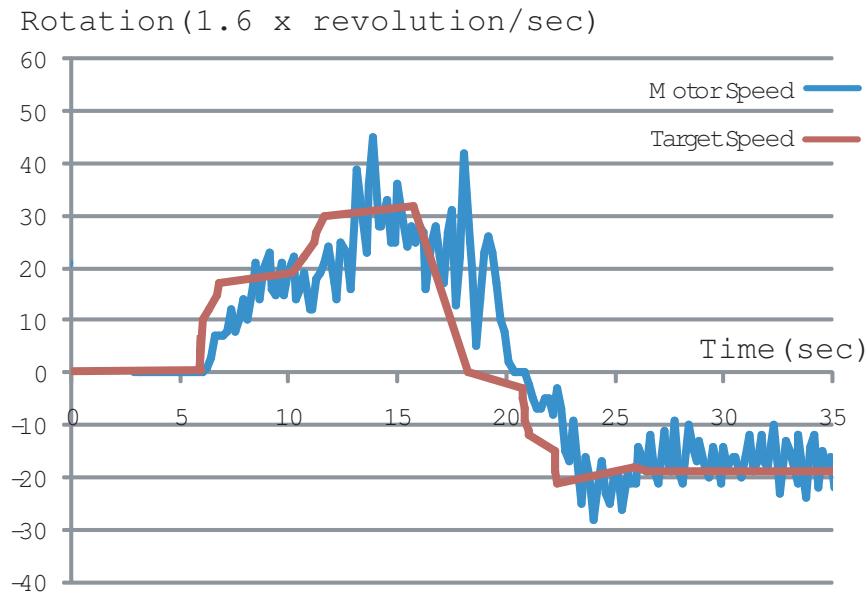


Figure 7.2: *Test result of motor speed control*

Figure 7.2 shows the test result of the speed control using the prototype robot. The smoother line shows the target speed, and the noisy line shows the measured motor speed. The measured motor speed did, in fact, follow the target speed with some time delay and some oscillation.

The speed control is similar to the "cruise mode", which is usually employed in high-grade automobiles. The cruise mode maintains the driving speed of a car by controlling the accelerator pedal. On the other hand, the voltage control, which is without any special control support, can be considered to be a normal driving way of simply adjusting the acceleration of an automobile. Therefore, it can be concluded that both the "speed control" and the "voltage control" can be implemented for spherical robots. The simple "voltage control" can be used for the normal driving with frequent breaks, acceleration, and turning. Alternatively, the "speed control" can be used for long distance cruising on a smooth terrain.

7.2 Roll angle stabilization

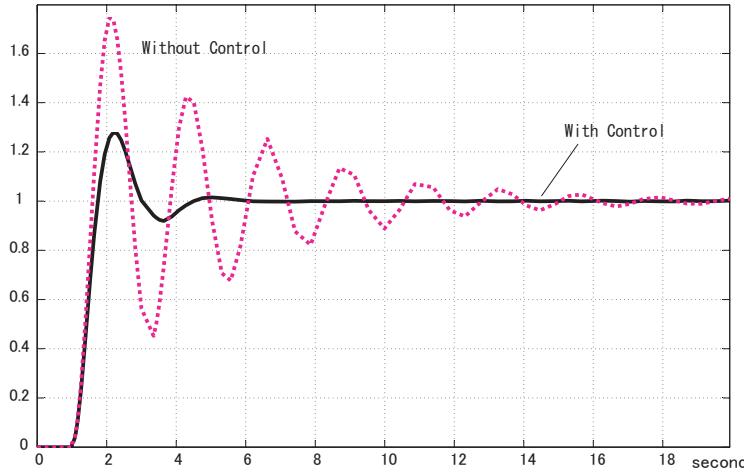


Figure 7.3: *Simulation result of roll angle stabilization*

Figure 7.3 shows the simulation result of the roll angle stabilization based on the modeling of the prototype. The model and parameters of this simulation are based on the block diagram shown in Figure 6.2 with the time delay of 0.05 seconds. The damping control is supposed to dampen the oscillation in a shorter time.

The characteristic equation of the roll motion can be easily derived from equation (6.6).

$$2.227 + 0.14s + 0.289 = 0 \quad (7.1)$$

$$\therefore s = -0.24 + 2.8i \quad (7.2)$$

Therefore, the natural frequency of this system is 2.8 (rad/s), and its period is 2.24 seconds, which can also be seen in Figure 7.5.

Figure 7.4 shows the testing result with the prototype robot. The period of its sinusoidal motion without stabilization is about 2.9 seconds, which is a bit different from the theoretically estimated period of 2.25 seconds. Nevertheless, The test results confirm the outcome of the simulation, dampening the oscillation in a shorter time. This stabilization makes the steering of the robot easy, since the roll angle of the robot, which determines the steering radius, can be stabilized.

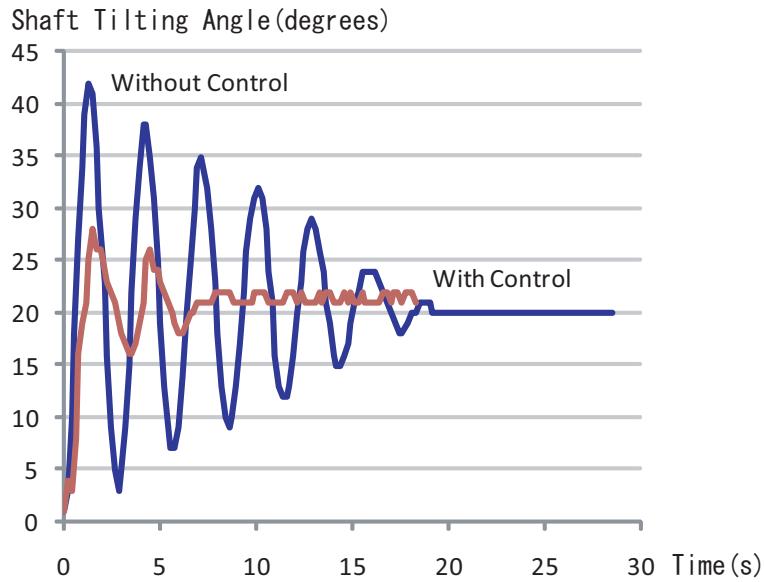


Figure 7.4: *Test result of roll angle stabilization*

7.3 Driving angle limitation

Driving angle limitation control also works well with the prototype robot – even if the ball gets stuck but still tries to accelerate, the pendulum does not swing up higher than a certain angle, thus avoiding the meaningless spinning of the pendulum.

7.4 Steering characteristics

Figure 7.5 shows the tests results and theoretical characteristics of steering. The three solid lines denote the theoretical characteristics derived from equation (3.25), and the dots denote the test results.

The test was conducted in the following way; the field was about 5 by 4 meters in size. Strips of tapes were stuck on the floor and marked red every 10cm for determining the location of the robot in the field. The motion of the robot was video recorded and, especially the steering motion, was analyzed afterward.

The rotational radius as well as the driving speed have been measured using the video, and plotted in Figure 7.5. The steering angle of the pendulum is

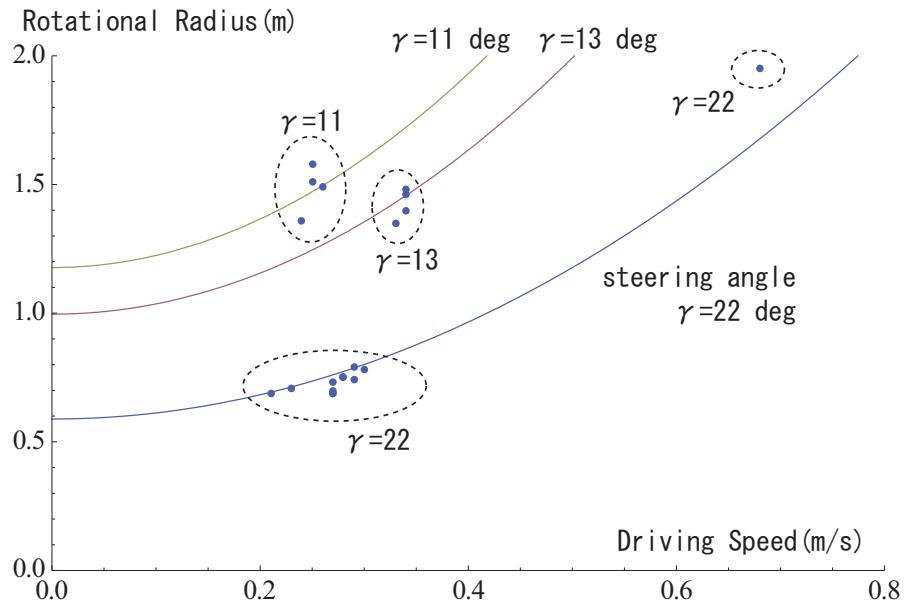


Figure 7.5: *Test results of steering*

determined by the commands from the GS, and also indicated in the figure. The results show that the theory and the hardware test almost match. Although the rotational radius has a certain amount of error from the theoretical value, the basic theoretical characteristics of steering has been proven to be correct.

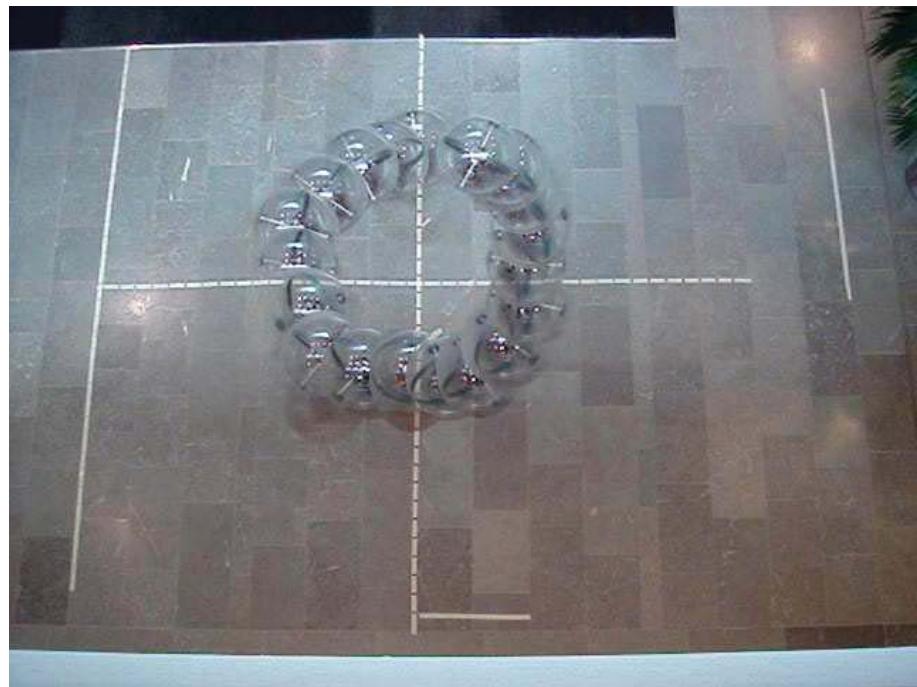


Figure 7.6: *Prototype robot and its trajectory*

Chapter 8

Conclusions

8.1 Prototype robot

The key features of the prototype robot come from the implementation of the newest technologies to streamline the research process. The following technologies have proven to be research-friendly during the study.

8.1.1 Bluetooth communication

One key technology used for the prototype robot is Bluetooth wireless connectivity. Bluetooth is a widely used commercial wireless system as well as being digital-friendly. Therefore, it is easy to implement in digital systems with a computer. In addition, the Bluetooth class-1 wireless device allows connectivity up to a range of 100m, which is more than adequate for this research purpose.

In addition, the Bluetooth-serial adapter used for the robot can be connected to the AVR easily, without any knowledge of networking which wireless LAN would require.

8.1.2 Low self-discharge Ni-MH battery

Low self-discharge Ni-MH batteries have two main features: low self discharge, as the name implies, as well as a reduced memory effect.

Low self-discharge characteristics allow the batteries to be ready for use even after several weeks, even more than a year, after charging. This feature reduces the frequency of charging especially when the robot is operated intermittently.

The other feature, reduced memory effect, makes the management of charging easy. Normal Ni-MH batteries should not be charged after being only partially discharged, since the memory effect lessens energy capacity. Therefore, the operator has to take into consideration the timing of charging for normal Ni-MH batteries, but does not have to worry when using LSD Ni-MH batteries.

8.2 Sensors for the feedback control

Some testing using the prototype robot revealed that it is not realistic to use differentiation of the measured data due to the noise. Therefore, the steering control definitely requires a gyroscope for the measurement of the angular rate of the roll angle of the robot. This research utilized the Kondo KRG-3 gyroscope attached to the main shaft to measure the angular rate of the roll angle of the robot.

It has also been proven that an accelerometer can be used to determine the pendulum tilting by measuring the gravity direction.

8.3 Theoretical analysis

8.3.1 Stability without sensors on the ball

Chapter 5 shows that the driving system is theoretically stable for any of the three inputs (voltage supply, motor speed, and motor torque input), even without any

sensors on the outer surface of the ball. This allows operation of the ball without sensors on its surface, thus avoiding the cabling problems between the sensors on the rotating sphere and the main board on the pendulum.

8.3.2 Gyro effect

Section 3.1 shows that with or without the gyro effect, the basic trend of the change in rotational radius has no change. Namely, the slower speed and the greater steering angle of the pendulum lead to steeper steering, despite the presence or absence of the gyro effect. In addition, the difference in rotational radius coming from the gyro effect is slight in the simulation speed range, meaning that the strategy for steering can remain the same.

8.4 Control methods

8.4.1 Stability

The fact that the robot is stable on its own is extremely important. As a result, the motor speed/torque control can be focused on without worrying about the ball stability.

8.4.2 Driving control

The input of the motor is the supply voltage. Therefore, voltage input is the most low-level input without any control support. Nevertheless, the voltage control can be used for normal operation with lots of braking, accelerating, and turning.

On the other hand, the speed control can be used for a cruise mode, where the speed should be maintained. This control can be realized with the simple PID control and the ultimate sensitivity method tuning. It has been proven to be working well from the test results.

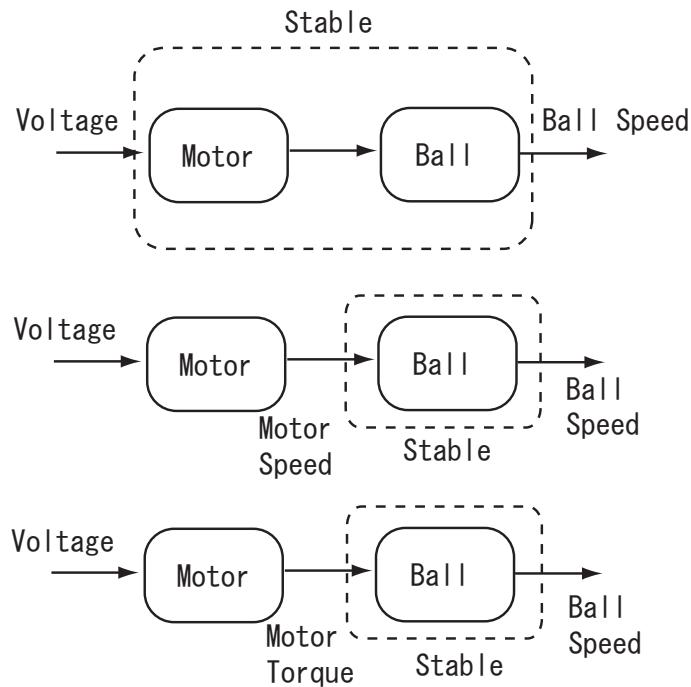


Figure 8.1: *Block diagram of the ball robot*

The torque control is, however, found to be unsuitable for supporting manual control support, since a smooth floor surface has only slight counter torque to cancel the input torque.

8.4.3 Steering control

The theoretical study (Chapter 3) shows that:

- Steering can be controlled by two input parameters: the traveling speed and the steering angle of the pendulum.
- Even if the gyro effect has a slight effect on the steering rate, the basic trend mentioned above remains the same.
- Rotational radius is also affected by the roll angle of the robot, which is not an input but one of the ball states. Therefore, the roll angle should be stabilized.

Consequently, Chapters 6 and 7 show that the roll angle can be stabilized with simple differential control.

8.4.4 Future works

Automated control This study focuses mainly on supporting control methods for manual control. Pure automated control without any manual operation can be implemented as a future study.

Servo motor synchronization The prototype robot uses two servo motors synchronized in order to control the pendulum sideways. However, the two motors are not always completely synchronized, thus wasting energy and wearing the gears of the motors trying to rotate in a opposite direction. Complete synchronization method should be somehow implemented in the future.

Localization Localization and navigation method of the robot can also be implemented, which is needed for pure automated control. More sensors for localization would be needed.

References

- CHEMEL, B., MUTSCHLER, E., AND SCHEMPF, H. (1999). *Cyclops: miniature robotic reconnaissance system. Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, 3.
- HALME, A. AND WANG, T. (1996). *Motion control of a spherical mobile robot. Advanced Motion Control, 1996. AMC'96-MIE. Proceedings., 1996 4th International Workshop on*, 1.
- JONES, J. (2001). *Inflatable robotics for planetary applications. 6th International Symposium on Artificial Intelligence, Robotics Automation in Space: A New Space Odyssey, Montreal, Canada, June 18, 2001.*
- KANO, M. (2005). *Control System Designing using MATLAB (MATLAB wo riyou shita seigyoiki sekkei)(Japanese).*
- URL:** <http://www-pse.cheme.kyoto-u.ac.jp/~kano/document/text-cont-basic.pdf>
- KNIGHT, W. (2005). *Spherical robot provides rolling security cover. NewScientist.com news service.*
- URL:** <http://www.newscientist.com/article.ns?id=dn6932>
- KOSHIYAMA, A. AND FUJII, K. (1996). *Development and Motion Control of the All-Direction Steering-Type Mobile Robot (4th Report, Mechanisms, Motion Principle, Control Methods and Experimental Results of a Stand-alone Spherical-wheeled Robot). Transactions of the Japan Society of Mechanical Engineers. C (Japanese)*, 62(602):3793–3801.
- KOSHIYAMA, A. AND YAMAFUJI, K. (1992a). *Development and Motion Control of the All Direction Steering-Type Mobile Robot (1st Report, A Concept*

- of Spherical Shaped Robot, Roll and Running Control). Transactions of the Japan Society of Mechanical Engineers. C (Japanese), 58(548):1128–1136.*
- KOSHIYAMA, A. AND YAMAFUJI, K. (1992b). *Development and Motion Control of the All-Direction Steering-Type Mobile Robot (2nd Report, Analyses and Experiments on Postural Stability and Ascent/Descent on a Slope. Transactions of the Japan Society of Mechanical Engineers. C (Japanese), 58(548):1137–1145.*
- KOSHIYAMA, A. AND YAMAFUJI, K. (1992c). *Development and Motion Control of the All-Direction Steering-Type Mobile Robot (3rd Report, Principle, Control Methods and Experiments on Steering of the Robot). Transactions of the Japan Society of Mechanical Engineers. C (Japanese), 58(548):1146–1153.*
- LAPLANTE, J.F. (2004). *Etude de la Dynamique D'un Robot Spherique et de son effet sur L'attention et la Mobilite de Jeunes Enfants.* (French).
- MICHAUD, F. AND CARON, S. (2000). *Roball - An autonomous toy-rolling robot. Proceedings of the Workshop on Interactive Robot Entertainment.*
- NAKAMURA, T. (2002). *Hikaru HALO, SONY ga kyuugata no robotto QTARO wo kaihatu. IT media news (Japanese).*
- URL:** <http://www.itmedia.co.jp/news/0203/27/qtaro.html>
- SPITZMUELLER, S. (1998). *Mircococontroller Based Control System for a Rolling Minirobot.* Master's thesis, Helsinki University of Technology.
- SUGIYAMA, Y., SHIOTSU, A., YAMANAKA, M., AND HIRAI, S. (2005). *Circular/Spherical Robots for Crawling and Jumping. Robotics and Automation, 2005. Proceedings of the 2005 IEEE International Conference on,* pages 3595–3600.
- TSAI, S., FERREIRA, E., AND PAREDIS, C. (1999). *Control of the Gyrover. A single-wheel gyroscopically stabilized robot. Intelligent Robots and Systems, 1999. IROS'99. Proceedings. 1999 IEEE/RSJ International Conference on,* 1.
- YLIKORPI, T. (2005). *A Biologically inspired rolling robot for planetary surface exploration.* Degree of licentiate of technology, Helsinki University of

Technology, Department of Automation and Systems Technology, Automation Technology Laboratory.

YOUNG, K. (2006). *Spherical micro-robots could explore Mars*. *NewScientist.com news service*.

URL: <http://space.newscientist.com/article/dn9610-spherical-microrobots-could-explore-mars.html>

Appendix A

Main Board Description

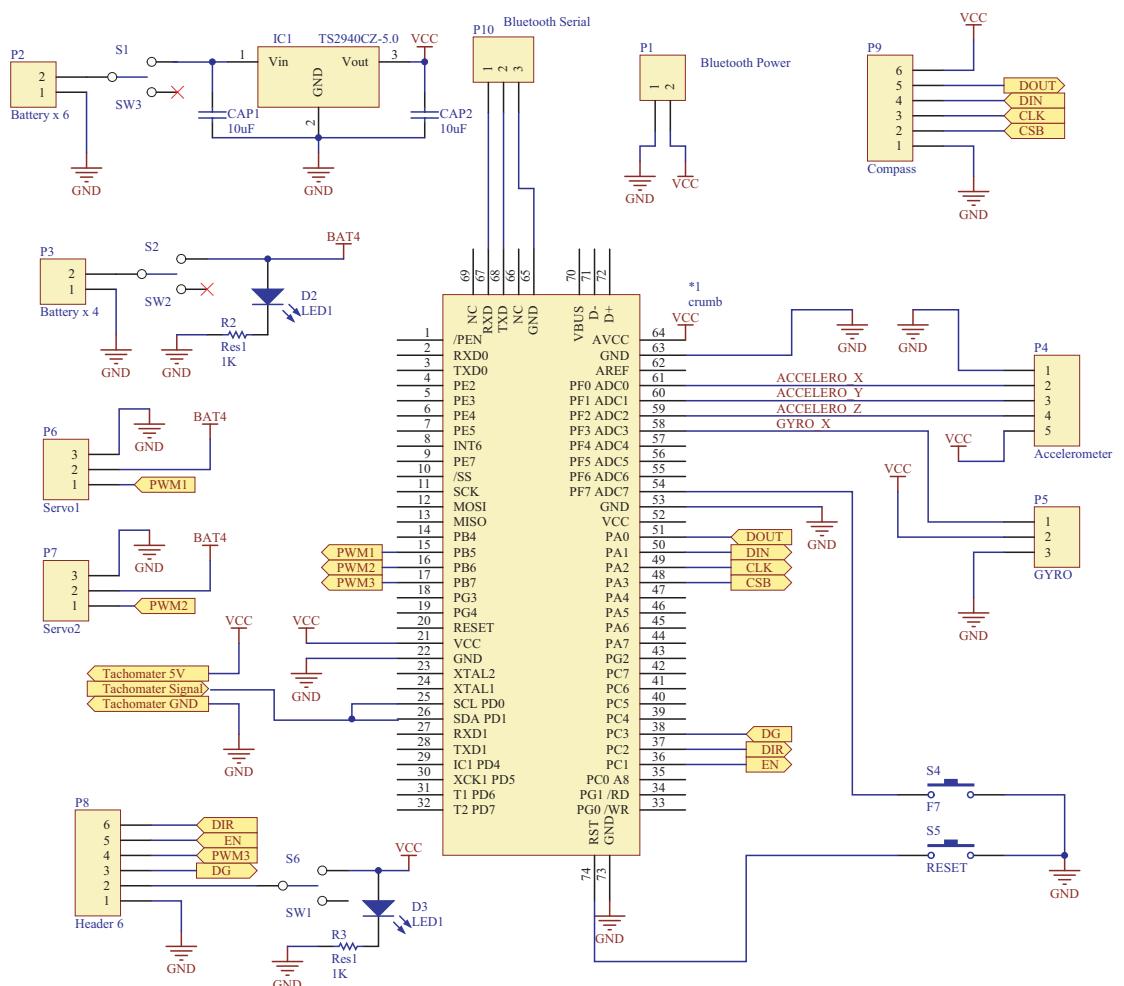
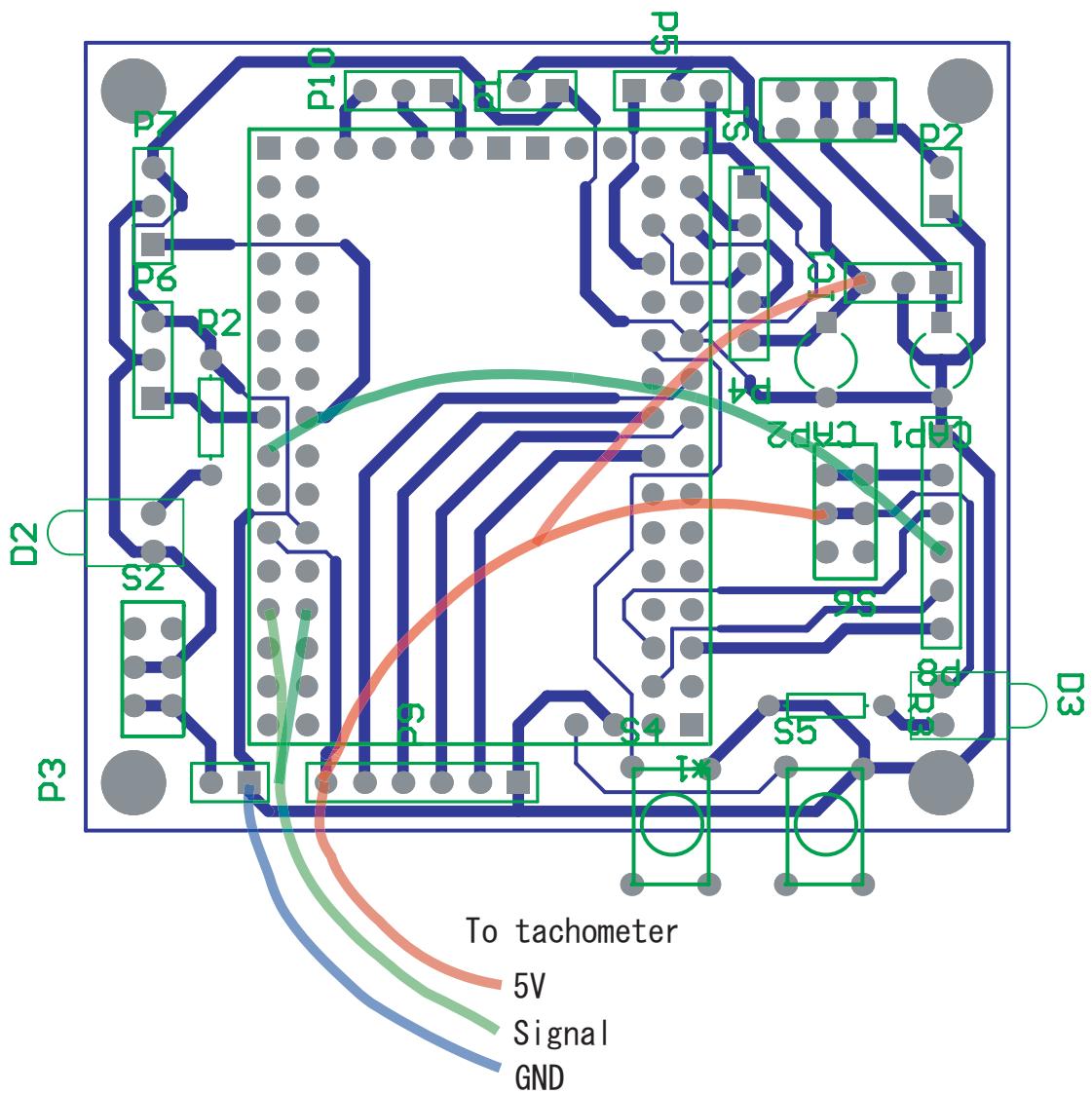


Figure A.1: Circuit design of the main board

Figure A.2: *Main Board Layout*

Component Name	Description	Location
AVRCrumb128	Main uchip	
4.7uF	Capacitor (electrolytic)	CAP1
0.01uF(103)	Capacitor (ceramic)	CAP2
LED	Servomotor ON/OFF indicator	D2
LED	Main board ON/OFF indicator	D3
L7805CV	5.0V Regulator	IC1
2-pin connector	5.0V for Bluetooth Adapter	P1
2-pin connector	Input Power Supply 7.2V	P2
2-pin connector	Input Power Supply 4.8V	P3
5-pin connector	For Accelerometer	P4
3-pin connector	For Gyroscope	P5
3-pin connector	For Servomotor1	P6
3-pin connector	For Servomotor2	P7
6-pin connector	For Motor Controller	P8
6-pin connector	For Magnetic Compass	P9
3-pin connector	For Bluetooth-Serial Adapter	P10
300	Resistor (metal-film)	R2
300	Resistor (metal-film)	R3
6-leg mini-switch	7.2V supply -> main board on/off	S1
6-leg mini-switch	Servomotor on/off	S2
Push-switch	not in use	S4
Push-switch	For reset	S5
6-leg mini-switch	5V from motor controller => main board on/off	S6

Table A.1: Component location on the main board

A Main Board Description

	Product Name	Manufacturer	Vender	Address of Vender	Qty.	Cost
Motors	Bluetooth-Serial Adapter	IOGEAR	eXpansys			72.38 EUR
	DC Motor RS540SH	Mabuchi (Japan)	Tsukumo	Tokyo, Japan	1	1280 JPY
	BMS660MG-HS Servo Motor	Bluebird (Taiwan)	Robot Factory	Osaka, Japan	2	8970 JPY
Sensors	Gyroscope Sensor KRG-3 (1 axis)	Kondo (Japan)	Tsukumo	Tokyo, Japan	1	5250 JPY
	3-axis accelerometer KXM52-1050		Akizuki	Tokyo, Japan	1	800 JPY
	Hitachi HM55B Compass Module	Parallelex inc.	ELFA		1	27 EUR
Battery	Photo Coupler (for optical encoder)				1	
	LSD-Ni-MH AA Batteries	Sanyo (Japan)	Edion	Mie, Japan	16	8370 JPY
	AVR Crumb board CAN		Chip45		1	39 EUR
Main Board	AVR Crumb board connector		Chip45		1	2.47 EUR
	Connectors	Nishikawa	Nishikawa	Tokyo, Japan	10	120 JPY
	6-leg switch	Akizuki	Tokyo, Japan		3	100 JPY
LED	Push-swtich				2	
	LED				2	
	Resister				2	
Capacitor	Capacitor				2	
	5V Regulator L7805CV				1	

Table A.2: Component list.

	Software	Purpose
Software development	Microsoft Visual C# 2008 Express Edition	Ground station software development
	Microsoft DirectX SDK (November, 2007)	Joystick implementation
	CodeVisionAVR (Ver. 1.24)	On board software development
Analysis	Mathematica 5	Dynamics modeling and analysis
	Matlab R2007a & Simulink	Control system designing and evaluation
Electronics	Altium Designer 6	Mainboard circuit and trax designing
	IVT BlueSoleil 6	Bluetooth connection software
Thesis writing	MiKTeX 2.7	Latex compiler
	TeXnicCenter 1 Beta 7.01	Latex editor
	dviout for Windows 3.08.2	DVI viewer
	JabRef 2.3.1	Reference management

Table A.3: Software list.

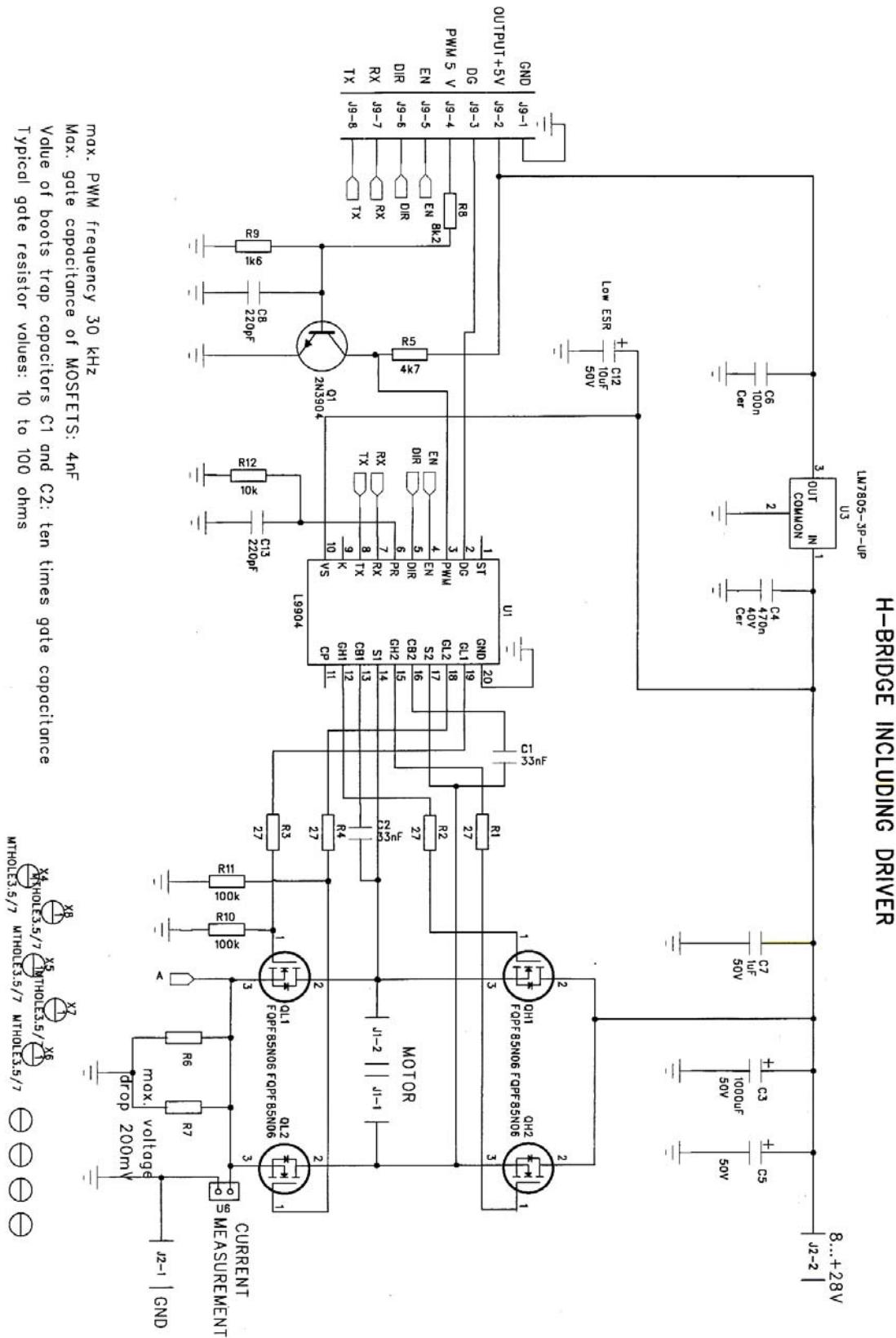


Figure A.3: Circuit design of the motor controller

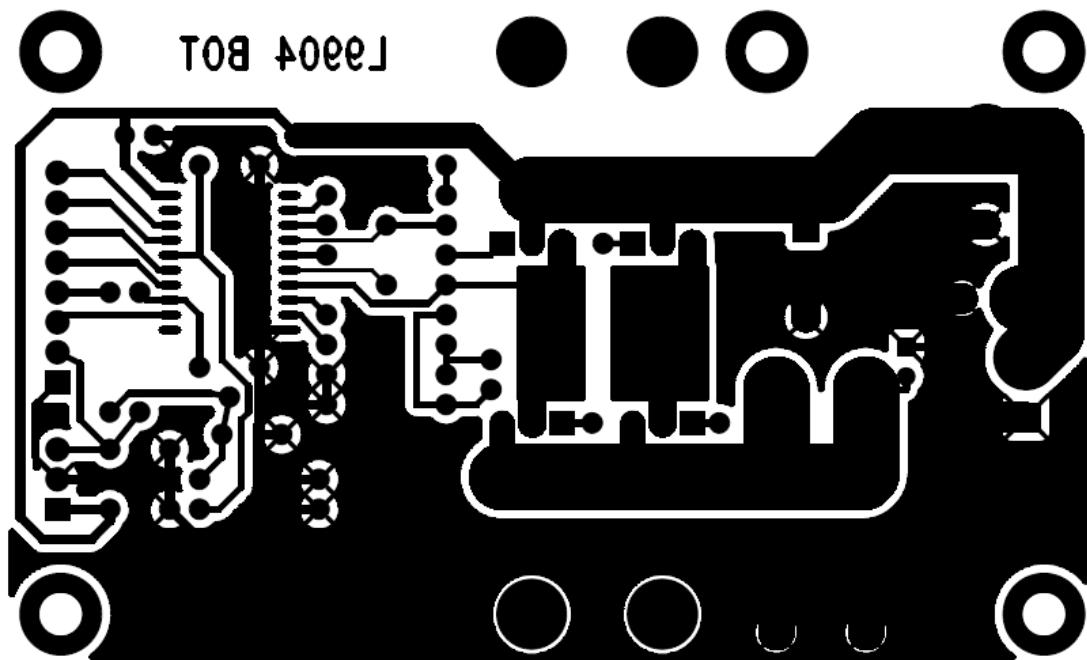


Figure A.4: *Motor controller trax layout*

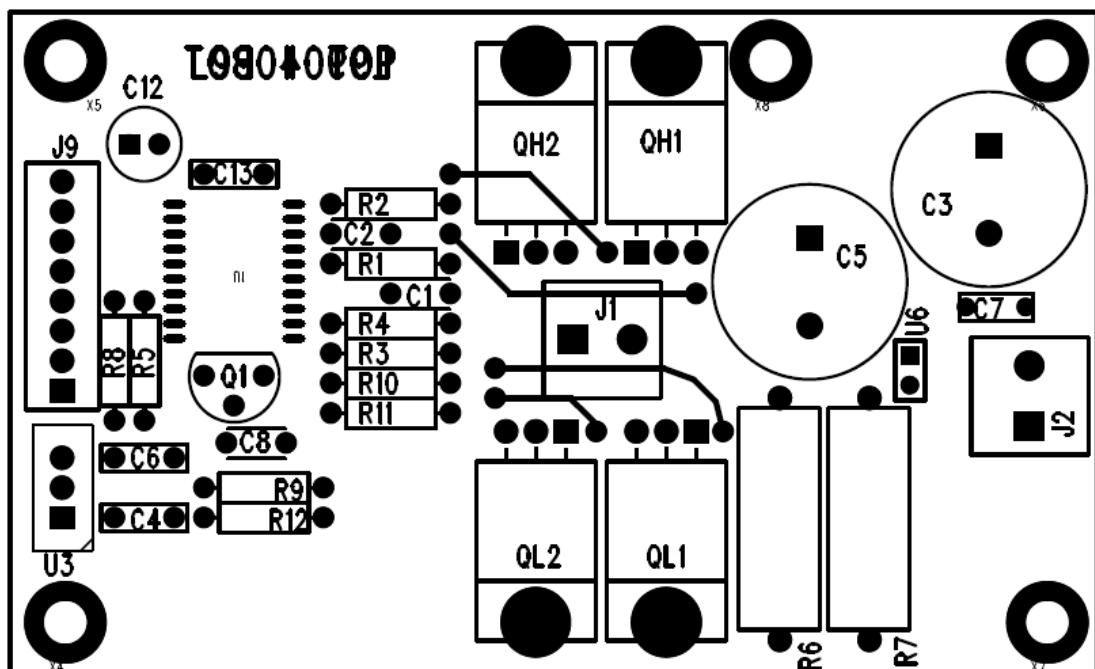


Figure A.5: *Motor controller assembly drawing*

Appendix B

Battery Charging

The charging procedure of the Ni-MH batteries used for the prototype robot is as simple as that of traditional Ni-MH batteries. Simply use the proprietary charger to charge the battery. The low self-discharge characteristics ease the charging management as follows:

- Charging after a slight discharge is possible without worrying about the memory effect.
- Long-term intermittent operation is possible without worrying about self-discharge. (Manufacturers claim that the batteries retain more than 85% of their capacity even one year after charging.)



Figure B.1: *SANYO Eneloop and its proprietary charger*

Appendix C

Prototype Robot Operation

This user instruction explains how to operate the prototype robot with the software *MarsBallController*, which was developed for the prototype robot operation. Here in this instruction, the prototype robot is referred as the *Mars Ball*.

C.1 Introduction

C.1.1 Definition, Benefits, and Purpose of the Procedure

MarsBallController is student-made software for controlling the spherical robot called Mars Ball, which is under development at the Automation and Systems Technology Department at TKK. The *MarsBall* is a new type of robot specially designed for Mars exploration.

MarsBallController enables the users to (1) control the *MarsBall* (driving and steering), (2) acquire data from the sensors installed on the *MarsBall*.

This procedure explains how to control *MarsBall* and acquire data from it using the *MarsBallController*.

C.1.2 Intended Audience

These instructions are written for students and researchers who want to use the *MarsBall* for their research or study in order to understand how they should use the *MarsBall* through the *MarsBallController*.

C.1.3 Prior Knowledge and Skills Needed by the Audience

To use the *MarsBallController*, you should already have some experience using the following:

- Bluetooth Management Software
- Windows OS (e.g., Click, Close, Double Click, Drag)

C.1.4 Brief Overall Description of the Procedure

To control the *MarsBall* robot, first you have to switch it on and connect your computer and *MarsBall* via Bluetooth wireless technology. In the second step, you have to start the *MarsBallController* software and connect it to the *MarsBall* through the Bluetooth connection established in the first step. In the third step, you can send commands to *MarsBall* to start, drive, acquire data, and stop. Finally, you need to close the connection and switch off the robot.

C.1.5 Materials, Equipment, and Special Conditions

Before you start controlling *MarsBall*, be sure you have the following items:

- *MarsBall* robot itself
- A windows computer equipped with Bluetooth wireless technology
- *MarsBallController* software (MarsBallController.exe file)

C.1.6 Working Definitions

MarsBall: A ball-shaped robot designed at the Automation and Systems Department at the Helsinki University of Technology for exploring the surface of Mars

MarsBallController: Software for controlling and acquiring data from the *MarsBall*

Bluetooth Wireless Technology: A specification for short-range radio links between mobile computers, mobile phones, digital cameras, and other portable devices

C.1.7 Warnings, Cautions, Dangers

- If you cannot connect the *MarsBall* with the Bluetooth wireless technology, uncoupling of the Bluetooth adapters might be needed before connection. Consult the instructions for your Bluetooth system , and try uncoupling your Bluetooth adapter first.
- To save the battery, be sure to switch off the robot after operation.
- Do not hit the *MarsBall* against walls. Do not operate *MarsBall* in crowded areas. Do not drop *MarsBall*, since the plastic outer body is fragile.

C.1.8 List of Major Procedures

1. Switching on *MarsBall*
2. Starting the *MarsBallController*
3. Connecting to *MarsBall* from your system
4. Controlling *MarsBall* with *MarsBallController*
5. Data Acquisition
6. Terminating Control of *MarsBall*
7. Disconnecting *MarsBall*

8. Shutting down *MarsBall*

C.2 Required Steps

C.2.1 Switching on *MarsBall*

1, Turn on the main switch on the side of the robot. A green light will appear on the electric main board.

C.2.2 Starting *MarsBallController*

2, Simply double click "MarsBallController.exe" on your Windows computer. A new window titled *MarsBallController* appears. Now you have successfully started *MarsBallController*.



Figure C.1: *MarsBallController Icon*

C.2.3 Connecting to *MarsBall* from your computer

You can connect to the *MarsBall* wirelessly using the Bluetooth wireless technology.

3, Run the Bluetooth management software on your Windows computer

4, Search for Bluetooth devices with the software. A Bluetooth device named *MarsBall* should appear.

5, Connect the Bluetooth device to the *MarsBall* using the Bluetooth management software. After successful connection, check the COM number for the Bluetooth



Figure C.2: *MarsBallController* window

connection. Now the connection between *MarsBall* and your system has been established.

C.2.4 Controlling the *MarsBall* with the *MarsBallController*

You can connect to the *MarsBall* using the connection established by the Bluetooth wireless technology mentioned above.

- 6, In the *Communication* groupbox, choose the COM port number that you just connected to via Bluetooth from the "COM port" Listbox.

7, In the *Communication* groupbox, click *Connect*. The textbox below the *COM port* Listbox turns green once you have successfully established a connection.



Figure C.3: *Communication Groupbox (Choose COM port and click Connect)*



Figure C.4: *Communication Groupbox after successful connection with the green textbox*

C.2.5 Connecting the Joystick (Option)

The *MarsBallController* is compatible with any USB Joystick or equivalent devices, such as a gamepad.

8, If you want to use a Joystick for controlling the *MarsBall*, connect the USB-Joystick device to any of your USB ports.

9, In the *Joystick* groupbox, click *Connect*. Upon successful detection and connection to the Joystick, the textbox below the *Connect* button will turn green.

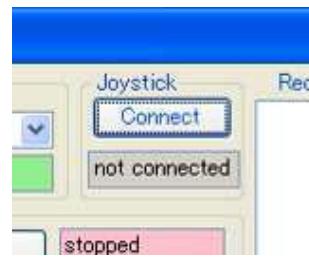


Figure C.5: *Joystick Groupbox*

C.2.6 Controlling *MarsBall*

You can control *MarsBall* either with the two Trackbars, one for driving forward/backward and the other is for steering, on the *MarsBallController* windows or with a Joystick if you have connected one.

10, In the *Control* groupbox, click *Start*. This sends a command for starting *MarsBall*. Once the controls become activated, the textbox in the top right corner of the group box turns green.

11, For controlling the speed of *MarsBall*, drag the *tick* in the *Driving Trackbar* left/right. The right side is for driving forward, and the left side is for moving backward. You can also control the speed and direction using a Joystick.

12, For controlling the steering of *MarsBall*, drag the *tick* in the *Steering Trackbar* left/right. The right side is for steering right, the left side is for steering left. You can also control the steering using a Joystick.

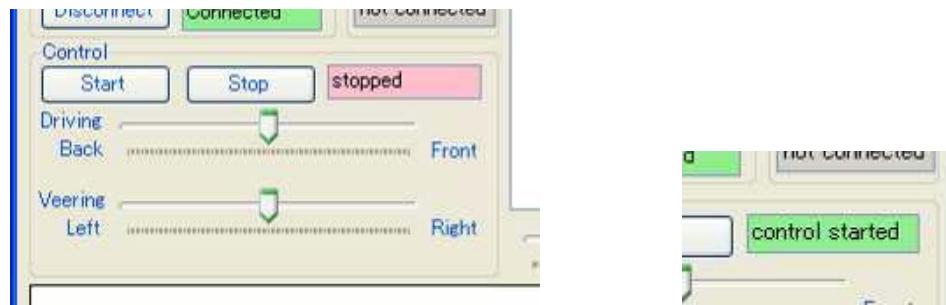


Figure C.6: *Control Groupbox* (Click *Start* first, then drag ticks in the trackbars)(Left side: before start controlling, Right side: after start controlling)

C.2.7 Data Acquisition

The four graphs at the bottom half of the *MarsBallController* show the following data in the time domain.(From above)

- Input Speed and Steering Level
 - Blue Line: Speed input by the user
 - Green Line: Steering angle input by the user
- Acceleration (triaxis) measured by the onboard accelerometer
- Angular rate measured by the onboard gyroscope
- Motor speed measured by the onboard optical encoder

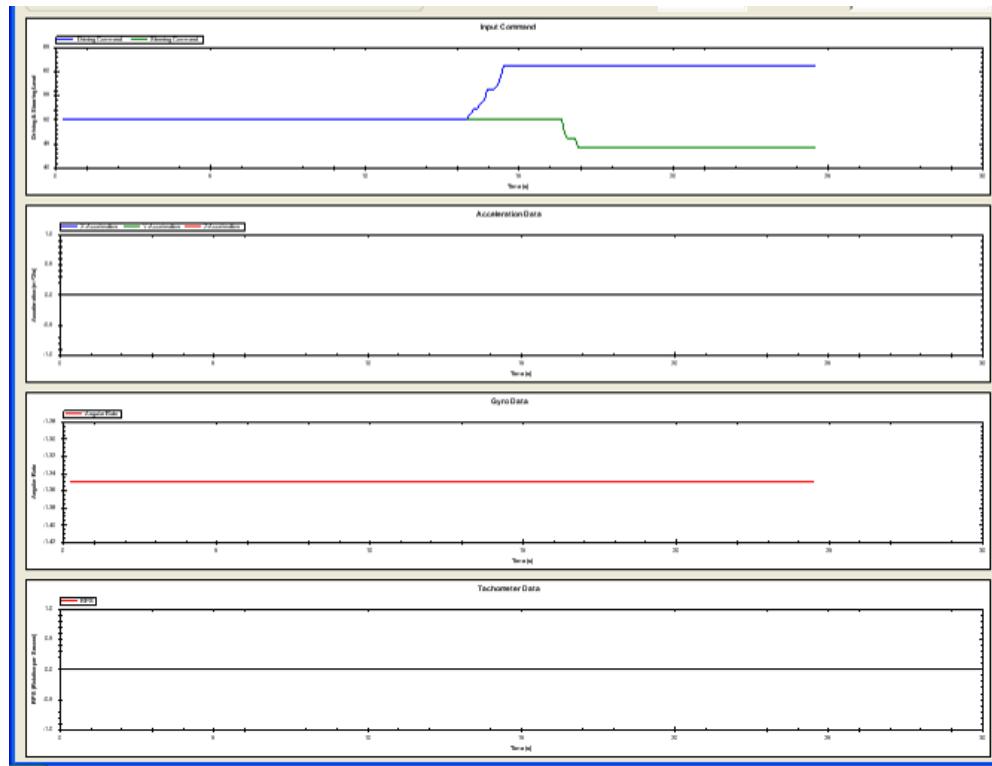


Figure C.7: Acquired data in graphs

The two rather large textboxes at the top-right of the *MarsBallController* show the transmitted/received data.

- The "Received Data Textbox" (left side) shows data sent from *MarsBall* to your computer in binary format. (Note: not in ASCII format) The data includes measurements by the onboard sensors.
- The "Command Textbox" (right side) shows data sent from your computer to *MarsBall* in binary format. (Note; not in ASCII format) The data includes commands to *MarsBall*, comprising the input speed and steering.

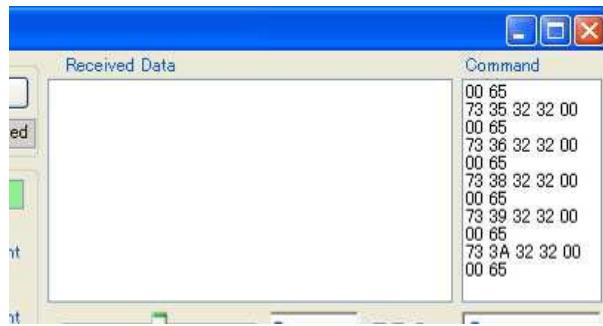


Figure C.8: *Received Data and Transmitted Command*

C.2.8 Terminating Control of *MarsBall*

13, Simply click *Disconnect* in the *Control* groupbox to terminate controlling. The *MarsBall* should stop moving within one second. The textbox in the top right corner of the groupbox turns red.

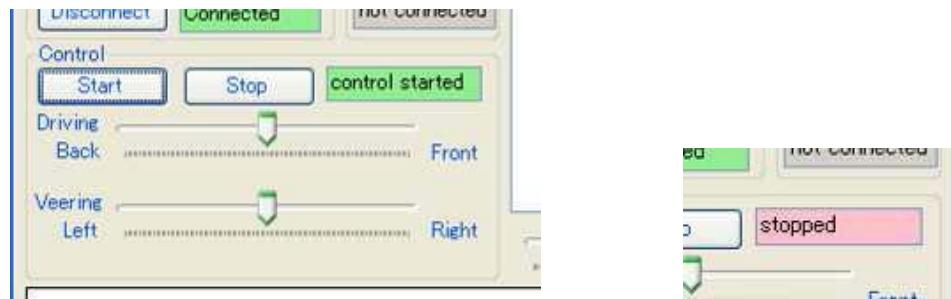


Figure C.9: *Control Groupbox (Click "Stop" to stop operation of MarsBall)*(Left: before stop controlling, Right: after stop controlling)

C.2.9 Disconnecting *MarsBall*

14, Simply click *Disconnect* to close the COM port.

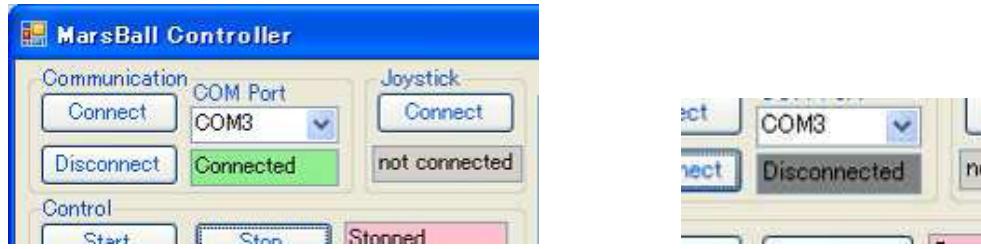


Figure C.10: *Communication Groupbox (Click Disconnect to disconnect and close the COM port)* (Left: before disconnection, Right: after disconnection)

C.2.10 Shutting down the *MarsBall*

15, Turn off the main switch of *MarsBall*. Otherwise, the battery will soon be discharged. The LED on the main board should turn off now.

Appendix D

Discussion about Rotational Radius

Sections 3.2.2 and 3.2.3 assume that the robot follows the circumference where the center point is the intersection of the floor and the line of the rotating shaft of the robot. This assumption can be written mathematically as:

$$\rho \tan \theta = r \quad (\text{D.1})$$

where ρ is the rotational radius of the traveling robot, θ is the steering angle of the robot, and r is the radius of the spherical robot. This assumption is theoretically proven in this appendix.

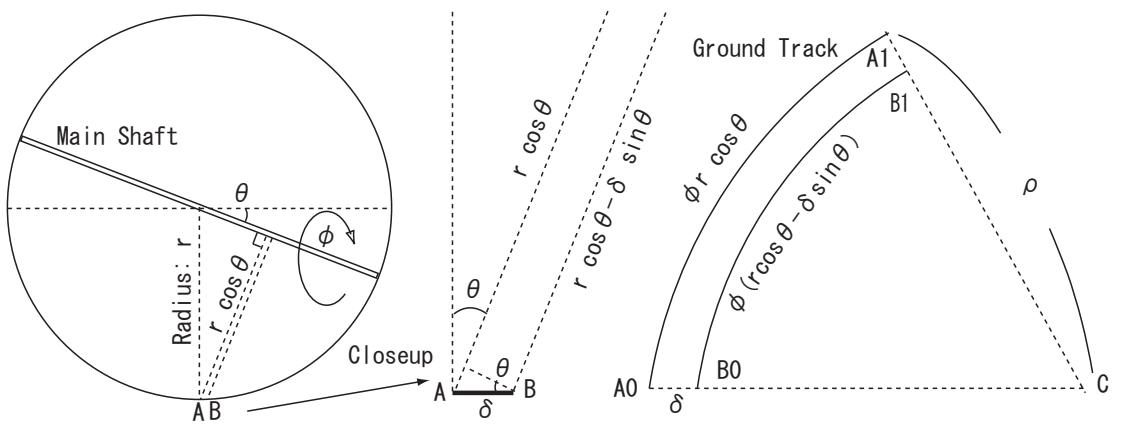


Figure D.1: *Spherical robot (left), closeup of between the robot and the surface (middle), ground track of the robot (right)*

First, it is assumed that the tiny line element AB is touching the surface that is driven over by the robot. (on the leftside of Figure D.1) Let θ be the steering angle of the robot, δ be the length of the element AB, r be the radius of the ball, and ϕ be the rotating angle of the ball.

Then, the distance from point A to the main shaft is $r \cos \theta$, and the distance from point B is $r \cos \theta - \delta \sin \theta$.

The rightside of Figure D.1 shows the ground track of the point A and B. The path length of point A and B are:

$$A_0 A_1 = \phi r \cos \theta \quad (D.2)$$

$$B_0 B_1 = \phi(r \cos \theta - \delta \sin \theta) \quad (D.3)$$

The following equations can be obtained regarding the angle $\angle A_0 C A_1$:

$$\angle A_0 C A_1 = \frac{\phi r \cos \theta}{\rho} \quad (D.4)$$

$$= \frac{\phi(r \cos \theta - \delta \sin \theta)}{\rho - \delta} \quad (D.5)$$

$$= \frac{\phi(\delta \sin \theta)}{\delta} = \phi \sin \theta \quad (D.6)$$

where ρ is the rotational radius of the ground track. (on the rightside of Figure D.1) Therefore, the rotational radius can be written as:

$$r = \rho \tan \theta \quad (D.7)$$

This equation proves that the assumption made in the steering motion analysis correct.

Appendix E

Friction Evaluation

This appendix shows that the friction between the prototype robot and the surface floor driven on avoids slippage of the robot. This no-slippage condition is assumed throughout this study.

E.1 Static friction coefficient

The website (<http://www.pk-shibuya.school-info.jp/fg-slope/RollingScience>) shows several friction coefficients between plastic and other materials as shown in table E.1.

Material 1	Material 2	Friction coefficient
plastic	wooden floor	0.57
plastic	rubber	1.72
plastic	cork	2.3
plastic	steel	0.38

Table E.1: Friction coefficients (static)

Although the material of the surface floor driven on by the robot is not fixed, it is assumed that the friction coefficient is more than 0.3 in this study, which is below the smallest value in the table.

E.2 Friction for driving

The friction coefficient of 0.3 allows the acceleration of maximum $9.8(m/s^2) \times 0.3 = 2.94(m/s^2)$ for the robot on a horizontal surface. This value can be interpreted as $2.94(m/s^2)/0.226(m) = 13.0(/s^2)$ of rotation acceleration $a_2 = \dot{\omega}_2$.

The simulation shown in Figure E.1 revealed that the rotation reaches acceleration of only $2.5(/s^2)$ at most, which is far below the possible maximum acceleration of $13.0(/s^2)$. This fact means that the robot has no slippage in its driving motion.

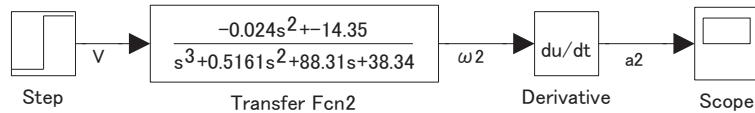


Figure E.1: *Simulation model for determining rotation acceleration of the robot*

E.3 Friction for steering

Using the equations (3.15) to (3.18), (3.25),(3.26), and (3.33), the friction force needed can be numerically calculated as shown in the following figure.

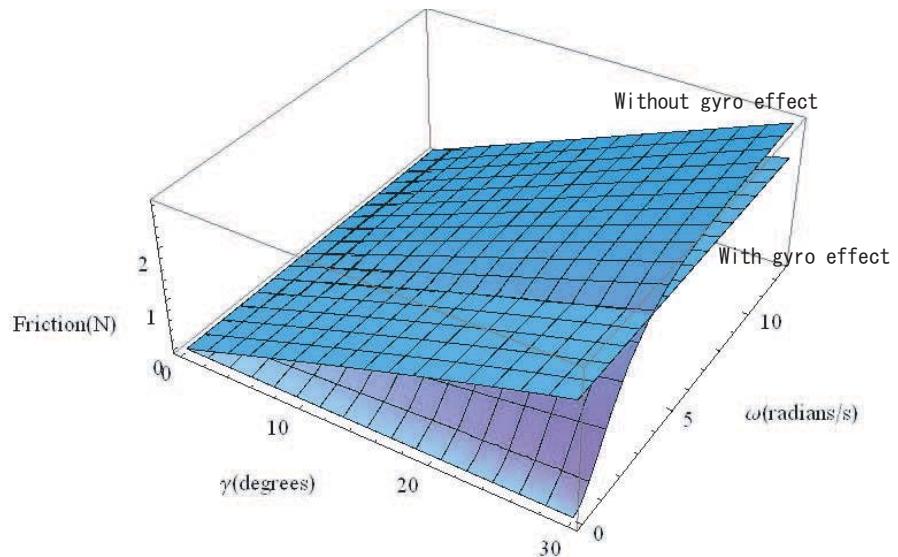


Figure E.2: *Friction force needed for steering*

The maximum friction force needed is less than 3N for the range of $\omega < 4\pi(radians/s)$

(rotation speed of the ball) and $0 < \gamma < 30$ (degrees) (steering angle of the pendulum).

This friction force is far less than the possible maximum static friction, which is:

$$\begin{aligned} & (\text{worst friction coefficient}) \times (M_1 + M_2) \\ &= 0.3 \times 5.089(kg) \\ &= 1.53(kgf) = 15.0(N) \end{aligned} \tag{E.1}$$

meaning that the robot can steer without slippage.