Motion Control of a Hexapod Robot Over Uneven Terrain Using Signed Distance Fields

by

Andries Phillipus Lotriet

Thesis presented in partial fulfilment of the requirements for the degree of Master of Engineering (Electronic) in the Faculty of Engineering at Stellenbosch University

Supervisor: Prof. J.A.A. Engelbrecht

March 2023

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: .... 2023/02/10 ....

Abstract

Motion Control of a Hexapod Robot Over Uneven Terrain Using Signed Distance Fields

A.P. Lotriet

Department of Electrical and Electronic Engineering,
Stellenbosch University,
Private Bag X1, Matieland 7602, South Africa.

Thesis: MEng (EE)

March 2023

In recent times great strides have been made in the field of autonomous robotics, especially with regards to autonomous navigation of wheeled and arial drones. Legged robotics however still face numerous problems before they can become practical to use, the most egregious of these problems being balancing of the robot and optimal foot placement.

This thesis focuses on providing a solution to the latter problem of foot placement. This is achieved by using an depth camera to, in real time, construct a localised map of the environment and subsequently analysing said map for optimal foot placement locations. The system is then tested using a hexapod robot both in simulation and on a physical robot.

Acknowledgments

Dedication

Table of contents

List of figures

List of tables

List of symbols

Constants

$L_0 =$	$300\,\mathrm{mm}$

Variables

Re_{D}	Reynolds number (diameter)	[ ]
x	Coordinate	[m]
$\ddot{x}$	Acceleration	[m/s^2]
θ	Rotation angle	[rad]
τ	Moment	[N·m]

Vectors and Tensors

$\overrightarrow{\boldsymbol{v}}$	Physical vector, see equation ...

Subscripts

a	Adiabatic
a	Coordinate

Abreviations

IK	Inverse Kinematics	vi
MuJoCo	Multi-Joint dynamics with Contact	2
GUI	Graphical User Interface	8
ROS	Robot Operating System	8
LiDAR	Light Detection and Ranging	7

Chapter 1

Introduction

1.1 Background

There are many applications where vehicles are required to traverse rough terrain, such as in mines, rescue operations, agriculture, construction, etc. In many of these use cases rough terrain makes the use of wheeled, or even tracked, vehicles difficult or impractical.

Compared to wheeled robots, legged robots could perform better in many of these environments, allowing navigation over terrain that would be impossible for wheeled or tracked vehicles to navigate. While legged robots possess extreme degrees of potential terrain traversability, advanced control and sensory systems are required to realise this potential.

1.2 Research Goal

The overarching goal of this project is to design and implement a sensory and control system that will allow a hexapod robot to autonomously walk over rough terrain.

This goal of the project is broken up into the following sub objectives:

1. Obtain a mathematical model of the robot, its actuators and its sensors.
2. Create a model of the robot in a simulation environment for development and testing.
3. Implement a vision based Simultaneous Localisation and Mapping (SLAM) system.

4. Develop a real time vision based dense mapping system for use in anchor point selection.
5. Develop a optimisation system to select optimal end effector anchor points based on the surrounding terrain.
6. Implement tilt stabilisation feedback control.
7. Implement and test the entire system on the physical hexapod robot.

1.3 Methodology

When deciding how to determine optimal end effector placement various sensing methods were considered, such as using a Red Green Blue Depth (RGB-D) camera to view the environment, placing force sensors on the robots end effectors or measuring servo torque to determine when the end effectors were in contact with a surface. A previous paper by Erasmus *et al.* (2023) used a RGB-D camera by storing past snapshots to adjust the end effectors to the optimal height, it was decided that the primary sensing method for this thesis would also be a RGB-D camera but instead of storing snapshots, a height map would be generated of the local environment. This would allow for more advanced methods of anchor point selection.

The first step in realising this system was to construct a accurate simulation of the hexapod. The primary simulation packages that were considered are Gazebo, PyBullet and Multi-Joint dynamics with Contact (MuJoCo). Gazebo was a appealing choice due to the easy integration with ROS, however it was decided to use MuJoCo since it was found to have superior contact physics simulation (Erez *et al.*, 2015).

Once the hexapod was adequately modelled in MuJoCo a tripod gait state machine, IK system and control interface was implement, at this stage the hexapod was capable of walking on flat terrain.

Next the the system to generate the height map was implemented, this entailed sampling the RGB-D camera and comparing cells in the height map against the depth buffer. Once the height map was implemented it was possible to build the system responsible for end effector placement, this is covered in detail in chapter 6, after which collision checking for the generated end effector motion was implement, ensuring that the hexapod does not get stuck on pieces of terrain.

With this the system was realised in simulation, next the system was implemented and tested on the physical robot, discussed in detail in chapter 7

1.4 Scope and Limitations

As the hardware used was developed by Erasmus *et al.* (2023) this project will focus only on developing the necessary software to control het robot hardware.

The velocity control, tilt angle stabilisation and end effector motion planner was developed by the author, while the low level IK controller used was developed by (Erasmus *et al.*, 2023). The scope of this project does not include autonomous waypoint navigation and thus requires a human operator to provide desired velocity commands. If no solution can be found for the given velocity command the system will not attempt to adjust the velocity command, the human operator will be required to adjust the command.

The local dense height map system was developed by the author, while the SLAM system used, ORB-SLAM3 was developed by Campos *et al.* (2021). It should be noted however that ORB-SLAM3 does generate a global sparse feature map of the environment, thus implementing waypoint navigation should be a trivial addition.

The sensors used in this project are limited to a single RGB-D camera, thus even with the generation of a local map, there could be cases where the system will not have height data around a desired anchor point. No torque or touch sensors are used to augment the system, thus if a leg were to collide with the terrain the robot will not adjust its trajectory. The system will however attempt to choose a step path based on the local heigh tmap such that no collision occurs. Additionally no Inertial Measuring Unit (IMU) is used, pose estimation is entirely handled by the SLAM system.

1.5 Thesis Outline

Chapter 2 provides a literature review on the methods of control, sensing and simulation used for hexapod robots.

Chapter 3 provides a overview of the hexapod hardware and the modelling thereof. This includes the robots mechanical form, sensors, on board computers and the simulation environment that is used.

Chapter 4 describes the environment mapping systems used, this includes the local dense height map and the sparse SLAM system.

Chapter 5 covers motion related topics, this includes the walking gait, IK and effector motion planning.

Chapter 6 describes the optimisation function and its various scores used to acquire the optimal end effector anchor points during each step taken.

Chapter 7 covers the hardware implementation process and software structure on the hardware.

Chapter 8 describes the various tests preformed and results obtained thereof.

Chapter 9 provides the conclusion of the research and any recommended future additions.

Chapter 2

Literature review

This chapter provides an overview of past research done regarding the control of hexapod movement and sensing methods. First a brief history of hexapods is presented after which various terrain sensing and adaptation methods are presented.

2.1 Hexapod history

Hexapoda, Greek for "six legs" refers the group of arthropods possessing three pairs of legs. As an example see a flesh-fly in Figure 2.1.

Figure 2.1: A Flesh-fly

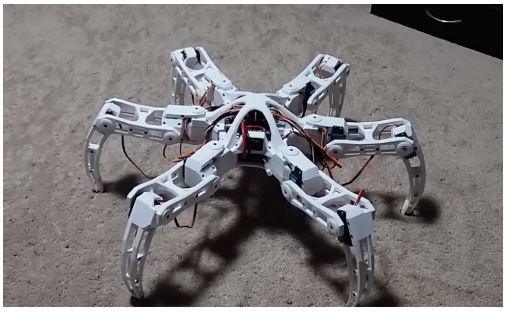

Figure 2.2: A circular hexapod

In the context of robotics "Hexapod" is used to refer to any robot with six legs, the most common configuration of Hexapods are either a rectangular layout with three legs on either side mimicking biological Hexapoda, or a circular design with radially symmetrical leg spacing, as seen in Figure 2.2

The hexapod possess the minimum number of legs to allow a naturally stable platform since while taking a step there can be upwards of three anchor points around the center of mass at all times. This makes the hexapod hexapods an ideal platform to navigate complex terrain while maintain stability, without requiring advanced balancing control systems.

For a hexapod to walk it must lift some of its legs while bracing with others, the number of swinging to bracing legs, and how each is moved, is referred to as the walking "gait". The chosen gait influences the speed and stability of the hexapod, the tripod gait is considered to be the most well rounded, having good speed and stability. In the tripod gait three legs are bracing while the remaining three swing. A example of a more stable gait would be the One by One gait, where only one leg is moved at a time.

It is also possible to create a system where there is no predetermined gait, but rather the system determines the optimal legs to brace and swing depending on the current walking environment.

2.2 Control

Walking over rough terrain requires a control system to correctly actuate the hexapods legs. Various types of control schemes exist, the primary schemes are traditional controllers, bio-inspired controllers and Reinforcement Learing (RL). These three schemes are discussed below. Control trends can be seen in Figure 2.3

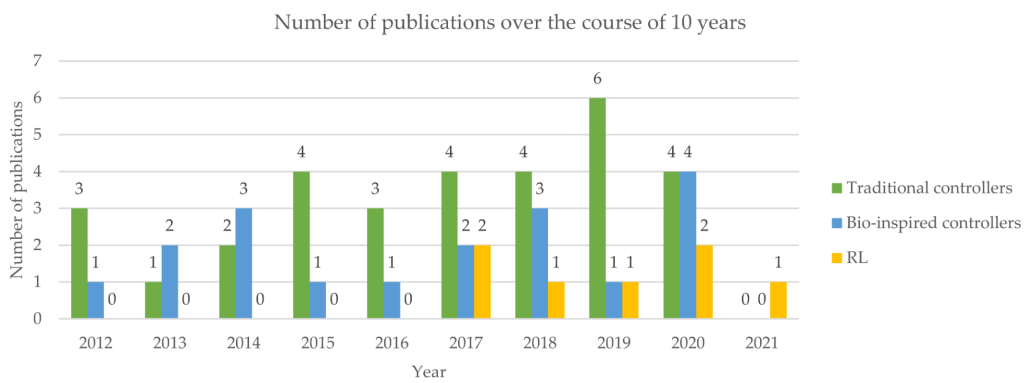

Figure 2.3: Trends of hexapod control schemes (Coelho *et al.*, 2021)

2.2.1 Traditional

Traditional controllers rely on an exact mathematical model of the robot and IK to calculate angular commands for all leg joints. This method of control is purely kinematic does not take into account external forces applied to the robot, thus it does not inherently adjust to the environment.

Instead of a purley kinematic model, a dynamic model can also be used. Using a dynamic model the forces acting on the robots legs are taken into account, usually acquired through torque measurements from servos. By taking applied torque into account dynamic model controllers will intrinsically detect a deviation when an external force is applied to the robot or its legs and compensate appropriately.

It should be noted that it is possible for a kinematic model controller to also adjust to external disturbances, but this is not intrinsic to the control model and requires additional control logic.

2.2.2 Bio Inspired

Bio inspired controllers attempt to mimic the neural structure of animals to achieve the same locomotion methods that they use. This is implemented through the use of a Artificial Neural Network (ANN) If implemented successfully a bio inspired controller can be highly adaptable to the surrounding environment and is even able to adapt to damaged or missing legs.

2.2.3 Reinforcement Learing

RL controllers are created through using trial and error to construct a neural net that minimises a cost function for a specific goal. This theoretically allows RL controllers to adapt to any circumstances given enough time, allowing a very hight level of autonomy, as no prior direction is required. RL controllers are though notoriously difficult to train properly, especially when the amount of sensors and control outputs grow large, increasing the feature space. And event he most well trained RL agent still has the possibility to exhibit inexplicable behaviour.

2.3 Sensors

No matter the control scheme used, to know where to place its feet the robot requires sensor(s) to sense its environment in some way, this could be achieved through simple sensors such as servo torque or touch, as used in **?**. Or more advanced methods such as vision or Light Detection and Ranging (LiDAR) could be used, as shown in **?**.

Depending on the terrain navigation system it might be required to localise the robot in 3D space, for this it is possible to use external sensors such as a type of beacon (RF, Reflective, Ultrasonic), Global Positioning System (GPS) or, through the use of a SLAM system, internal sensors, such as vision could be used.

The

2.3.1 End Effector Placement Method

Among other research focused on hexapods, many focus on topics such as obstacle avoidance, climbing surfaces, confined surfaces and cargo transportation. When focusing of terrain adaptation most often the use of sensors such as LiDAR, torque, or touch are employed. Where usually the height of end effectors are adjusted to the height of the terrain Coelho *et al.* (2021).

Some papers, such as Homberger *et al.* (2017) utilise stereoscopic vision, in addition to end effector height adjustment, also focus on surface material classifications based on which the virtual stiffness of the impedance controller is adjusted.

The focus of this paper will be on end effector height and planar position adaptation through real time walkability classification of the environment. While only utilising an RGB-D camera as sensor

2.3.2 Localisation and Mapping

This project requires a system that will localise the robot within its environment, as the primary sensor used is an RGB-D camera various visual SLAM systems were considered. ORB-SLAM 3, a optimisation-based, sparse map SLAM system was chosen to be used. ORB-SLAM 3 maintains a sparse map, an atlas, of both active and dormant features. This atlas is used to localise in the sparse map (Macario Barros *et al.*, 2022).

The implementation of a dense map to be used for end effector placement is discussed in chapter 4.

2.4 Simulation Environment

The most popular physics simulators for robotics in recent times are Gazebo, MuJoCo and CoppeliaSim (previously V-REP) (Collins *et al.*, 2021). Gazebo and CoppeliaSim both have easy to use Graphical User Interface (GUI) interfaces and easy integration with Robot Operating System (ROS). MuJoCo on the other hand does not have a full GUI interface, only a simulation viewer, and does not have native ROS integration. Having said this MuJoCo was found to be the most accurate and fastest simulator when considering the use case of robotics (Erez *et al.*, 2015).

Considering that the only relevant downside to MuJoCo is the lack of native ROS integration and the lack of a comprehensive GUI, which seeing as MuJoCo has good python bindings, could be seen as a advantage, MuJoCo was chosen as the simulator.

Chapter 3

Modelling

This chapter covers the simulation environment and modelling of the hexapod in MuJoCo.

3.1 Hexapod Construction

3.2 Servo Modelling

3.3 Simulation Terrain

Chapter 4

Mapping

In this chapter the localisation and mapping systems will be covered.

4.1 Localisation and sparse map

4.2 Dense map

Chapter 5

Motion

This chapter describes the systems governing the motion of the robot, such as leg motion planning and gait generation.

5.1 Gait State Machine

5.2 Inverse Kinematics (IK)

5.3 Foot Motion

5.4 Foot Placement

Chapter 6

End Effector Placement

A terrain scoring and optimisation function that executes on the height map is used to optimise the anchor points of the three supporting end effectors. This chapter covers the optimisation function and parameters used for this.

6.1 Scoring

6.1.1 Terrain Proximity

6.1.2 Slope

6.1.3 Height Delta

6.2 Placement Optimisation

6.2.1 Cost Function

6.2.2 Optimisation Function

Chapter 7

Hardware Implementation

This chapter describes the process of implementing the system built in previous chapters on the physical robot.

Chapter 8

Testing

This chapter covers all tests performed to validate performance of the system.

Chapter 9

Conclusions

Appendix A

Mathematical proofs

A.1 Euler's equation

Euler's equation gives the relationship between the trigonometric functions and the complex exponential function.

$$e^{i\theta} = \cos\theta + i\sin\theta \tag{A.1}$$

Inserting $\theta = \pi$ in (A.1) results in Euler's identity

$$e^{i\pi} + 1 = 0 \tag{A.2}$$

A.2 Navier Stokes equation

The Navier–Stokes equations mathematically express momentum balance and conservation of mass for Newtonian fluids. Navier-Stokes equations using tensor notation:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}\left[\rho u_j\right] = 0 \tag{A.3a}$$

$$\frac{\partial}{\partial t}\left(\rho u_i\right) + \frac{\partial}{\partial x_j}\left[\rho u_i u_j + p\delta_{ij} - \tau_{ji}\right] = 0, \quad i = 1, 2, 3 \tag{A.3b}$$

$$\frac{\partial}{\partial t}\left(\rho e_0\right) + \frac{\partial}{\partial x_j}\left[\rho u_j e_0 + u_j p + q_j - u_i \tau_{ij}\right] = 0 \tag{A.3c}$$

Appendix B

Experimental results

List of references

Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M. and Tardós, J.D. (2021). Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890.

Coelho, J., Ribeiro, F., Dias, B., Lopes, G. and Flores, P. (2021). Trends in the control of hexapod robots: a survey. *Robotics*, vol. 10, no. 3, p. 100.

Collins, J., Chand, S., Vanderkop, A. and Howard, D. (2021). A review of physics simulators for robotic applications. *IEEE Access*, vol. 9, pp. 51416–51431.

Erasmus, S. *et al.* (2023). Guidance, control, and motion planning for a hexapod robot moving over uneven terrain.

Erez, T., Tassa, Y. and Todorov, E. (2015). Simulation tools for model-based robotics: Comparison of bullet, havok, mujoco, ode and physx. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4397–4404.

Homberger, T., Bjelonic, M., Kottege, N. and Borges, P.V. (2017). Terrain-dependant control of hexapod robots using vision. In: *2016 International Symposium on Experimental Robotics*, pp. 92–102. Springer.

Macario Barros, A., Michel, M., Moline, Y., Corre, G. and Carrel, F. (2022). A comprehensive survey of visual slam algorithms. *Robotics*, vol. 11, no. 1, p. 24.