

# A Novel Design of a Salamander Robot Based on 3D Skeleton Kinematics of Real Salamanders

NAVID KHAJEH MAHABAD



**KTH Computer Science  
and Communication**

Master of Science Thesis  
Stockholm, Sweden 2011

# A Novel Design of a Salamander Robot Based on 3D Skeleton Kinematics of Real Salamanders

NAVID KHAJEH MAHABADI

Master's Thesis in Computer Science (30 ECTS credits)  
at the Systems, Control and Robotics Master's Program  
Royal Institute of Technology year 2011  
Supervisor at CSC was Örjan Ekeberg  
Examiner was Anders Lansner

TRITA-CSC-E 2011:136  
ISRN-KTH/CSC/E--11/136--SE  
ISSN-1653-5715

Royal Institute of Technology  
*School of Computer Science and Communication*

**KTH CSC**  
SE-100 44 Stockholm, Sweden

URL: [www.kth.se/csc](http://www.kth.se/csc)

# Abstract

This project addresses the design and development of an amphibious salamander robot based on the 3D kinematic data obtained from the skeleton of a real salamander. To date, studies on kinematic data have used conventional methods (e.g. markers on body) to track the body movements while focusing either on limbs or the spine. However, no study has concentrated on the kinematics of the skeleton to investigate the entire body movements in 3D by means of recorded X-ray cinematography.

There are a few prototypes of salamander robots developed so far. These robots are neither designed based on the accurate kinematic data nor optimized in terms of the joints. In addition, they lack several degrees of freedom to increase the controllability and represent a closer model of the real salamanders. Furthermore, there is only one amphibious robot that could swim and walk.

In this project a novel salamander-like robot was developed based on the following methodology. The kinematic data was collected from recorded X-ray videos of a real salamander. Then, necessary software tools were developed to accurately analyze the kinematic data. An optimization problem was solved to find the number and positions of the optimal joints in the spine. Based on the optimization results and also modeling simplifications, we created a model of the robot and verified the model by simulation. Finally we designed, constructed and controlled the robot to evaluate the performance of the model in practice. Although the robot was designed to address both terrestrial and aquatic locomotion, the focus of this project was on the terrestrial locomotion. The robot could be further developed to address the aquatic part (e.g. by waterproofing) which is out of the scope of this project.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Previous Studies on Salamander Locomotion . . . . .	1
1.2	Related work . . . . .	3
1.3	Project Goals and Objectives . . . . .	3
1.3.1	Summary of Report . . . . .	4
<b>2</b>	<b>Kinematic Data Collection</b>	<b>7</b>
2.1	Different Methods of kinematic data collection: . . . . .	7
2.2	Data Collection from X-ray Videos . . . . .	9
2.3	Data Conversion and Calibration . . . . .	11
2.3.1	Calibration Methodology . . . . .	11
2.3.2	Conversion Tips . . . . .	13
<b>3</b>	<b>Kinematic Data Analysis</b>	<b>15</b>
3.1	Software Development for Data Analysis . . . . .	15
3.2	Definition of Variables . . . . .	17
3.3	Results . . . . .	18
3.4	Discussion . . . . .	21
<b>4</b>	<b>Model Optimization and Simplification</b>	<b>25</b>
4.1	Spine Joint Optimization . . . . .	26
4.1.1	Spine Optimization - Analytical Approach . . . . .	27
4.1.2	Spine Optimization - Numerical Approach . . . . .	29
4.1.3	Spine Optimization Results . . . . .	30
4.2	Limb Model Simplification . . . . .	32
4.2.1	Hindlimb Forward Kinematics . . . . .	34
4.2.2	Limb Model Simplification Results . . . . .	37
<b>5</b>	<b>Mechanical Design and Development of the Robot</b>	<b>39</b>
5.1	Design Tips and Criteria . . . . .	39
5.2	Spine Design . . . . .	41
5.3	Hindlimb Design . . . . .	43
5.4	Forelimb Design . . . . .	46
5.5	Robot Development . . . . .	47

<b>6 Physical Simulation</b>	<b>51</b>
6.1 Model Simulation in Webots . . . . .	52
6.1.1 Results . . . . .	53
<b>7 Robot's Hardware and Control</b>	<b>55</b>
7.1 Robot's Actuators . . . . .	55
7.2 Robot's Control Hardware . . . . .	56
7.3 Robot Control . . . . .	58
7.3.1 Angles and Definitions . . . . .	59
7.3.2 Kinematic Calculations for Robot Control . . . . .	61
<b>8 Discussion</b>	<b>67</b>
<b>9 Conclusion Remarks and Future Works</b>	<b>69</b>
<b>Bibliography</b>	<b>71</b>

# **Chapter 1**

## **Introduction**

Locomotion is a fundamental capability for animals to interact with their environment, search for food, nest, and survive. In the study of the animals locomotion, salamanders are crucial, since among vertebrates, they are the closest animals to early tetrapods in terms of movement patterns and morphology [24], [19]. In addition, it is believed that the evolutionary changes of salamanders remained intact with regards to their morphology which leads to many similar features with early tetrapods [24]. These features make the salamanders an important subject for sprawling locomotion studies.

Furthermore, as an amphibian, salamanders are regarded as a key animal from which the evolutionary changes from aquatic to terrestrial locomotion can be inferred [7]. They are capable of walking on the ground as well as in water when submerged [24], [4] and also swimming [22]. Therefore, salamanders can provide an interesting model for the locomotion studies of the transition from water to land. In addition, the static morphological evolution of salamanders during the past 150 million years draws the attentions of scientists as a remarkable field of study [4], [19] and [21].

In general, salamanders exhibit a sprawling posture in terrestrial locomotion in which the limbs (forelimbs and hindlimb) are out from the body and placed in the sides. In contrast to the erect posture where the balance is obtained over the feet, the sprawling posture does not employ balance and equilibrium is not an issue. Salamanders also employ axial undulation of spine to propel through the water in the aquatic locomotion.

### **1.1 Previous Studies on Salamander Locomotion**

Previous studies of salamander locomotion have focused on the limbs or axial movements of the body to categorize different gaits. Some studies such as [23] and [24] have focused on the limb kinematics of salamanders in terrestrial locomotion. In addition, backward and forward kinematics of hindlimbs have been investigated in [25] during walking. Instead, other studies have focused on the kinematics of axial

## CHAPTER 1. INTRODUCTION

movements ([14], [22] and [13]). Other researchers have tried to make a biomechanical model of salamanders in which an amphibious robot inspired from salamanders is developed to study their locomotion ([18], [2], [16], and [17]). Such studies have tried to understand the neural mechanism and circuitry which contribute to controlling salamander locomotion. A 3D musculo-mechanical model of salamanders has been introduced by [6] to investigate the locomotion of walking on ground and in water. In this model the legs played the role of holding the body to avoid touching the ground and they were capable of lateral movements. Furthermore, the model has been used to investigate the walking and trotting gaits and turning gaits involving trunk bending.

On the other hand, other studies focused on the muscles electromyographic (EMG) activities to describe the kinematics. Patterns of epaxial muscle activation along the trunk have been investigated in [22] and [8] in swimming and walking gaits by means on EMG recording. In addition, [8] has analyzed the patterns of several limb muscles. [25] has assessed EMG recorded data from the hindlimb muscles during forward and backward locomotion while combining this data with the kinematics of the body movements. In this study, in addition to the EMG data, the movements of the body had been recorded by high speed cameras. The body movements were monitored by painting some parts of body (hindlimbs and trunk) with white dots. [27] has also studied the activities of seven trunk muscles with EMG recording during swimming, trotting in the water and on the ground.

Salamanders are known to have symmetric gaits [14]. Unlike mammal, salamanders lack support by limbs under the body (erect posture). This explains why they cannot use asymmetric gaits as quadrupeds with sprawling posture [23].

As amphibians, salamanders employ different gaits during aquatic and terrestrial locomotion. To propel through the water, they use an anguilliform gait which a traveling wave propagates in the body from head to tail [22], [2]. In this gait, the limbs are tonically activated and kept folded close to the body without assisting locomotion.

Another gait that salamanders use in water is underwater stepping when they are submerged. [24] has studies the kinematics of submerged walking and discussed the function of the limbs as a lateral sequence walk to propel.

On the ground, salamanders employ walking and trotting gaits. In contrast to the swimming gait, during walking, the body creates an S shape standing wave in the spine [22], [23], [8] and [18]. In this gait, the limbs play an important role and they are coordinated with the bending of the spine to form a lateral sequence walk [15] (i.e. for a given hindlimb footfall, the next footfall is the forelimb in the same side, and footfalls of diagonal limbs are close in terms of timing). Salamanders also employ trotting gait which could be categorized according to the limbs footfalls. In trotting, the diagonal limbs are approximately in synchrony [15]. [14] stated that in this gait the body tends to produce traveling waves, however [23] has discussed patterns of both traveling and standing waves in the body while trotting.

In this project the walking gait of salamanders will be studied by investigating the kinematic data obtained from recorded X-ray videos. The videos represent the

## 1.2 RELATED WORK

walking gait of the salamanders which will be used to extract the kinematic data of the skeleton in 3D. In the following chapters, the method used to collect the data will be discussed.

### 1.2 Related work

There are few prototypes of salamander robots developed so far. "Salamandra robotica" [17] is a salamander-like robot inspired from real salamanders. The robot is used as a tools to test biological hypotheses concerning the locomotion. The 85-cm-long robot contains 6 actuated joints in the spine and 2 joints in the limbs. Unlike real salamanders which exhibit complex limb structure, the limbs in this robot only employ continuous rotation. As an amphibious, the robot is capable of robust walking, swimming and turning (on the ground and in water). The robot is controlled by the concept of central pattern generator (CPG) and the controller accompanied with the mechanics lead to a robust locomotion of the robot.

The new version of "Salamandra robotica" robot is "Salamandra robotica II" which is also an amphibious robot. New limb design, passive tail and using more degrees of freedom in the spine are the improvements made to this robot [20].

"Robo-Salamander" [5], is a robot prototype developed preliminary as a platform for biological research. The robot which is only able to work on the ground, contains 2 motors on the spine, 1 for the lateral bending of the spine and 1 for longitudinal rotation. As a four-legged crawler, the robot uses 2 motors for each leg. No experiments with this robot have been reported, and no other publications followed.

Another salamander robot is a robot with 6 segments presented in [1]. The hardware of the robot is based on FPGA and the robot is not an amphibious and hence can only walk.

There are also some legged robots with flexible spine that have been developed. These robots are not built as a platform to address the scientific experiments but as toys for entertainment.

Only a few salamander like robots (e.g. Salamandra robotica) exist in publications as a platform of scientific studies. A good model of a salamander as an articulated body, should employ several degrees of freedom especially in the spine. Such a robot has not yet been developed. Apart from "Salamandra robotica", none of these robots are capable of swimming. None of these robots are designed based on the kinematic data, nor optimized in terms of number and positions of the joints.

### 1.3 Project Goals and Objectives

As mentioned earlier, there exists a few prototype of robots that could potentially be used as a scientific platform to describe the salamander kinematics and locomotion. To date, no attempt has been made to describe the movements of the various joints of the salamander skeleton (including limbs and spine) in detail. In order to

scrutinize the kinematics of different gaits, X-ray cinematography has been used to track the bones and joints involved in the salamander locomotion in 3D.

In this project, the salamander's locomotion for ground stepping is thoroughly studied. The main goal of the project is to study the kinematics of the real salamander and accordingly develop a novel salamander-like robot which possesses several degrees of freedom to represent a realistic model of the real salamander. According to the kinematics of a salamander skeleton, and also the optimization of the joints, the robot will be designed. The robot locomotion is supposed to reproduce the walking gaits that are recorded from X-ray. The methodology used for this project starts from data collection, and analysis, then it continues to model optimization, design and simulation, and it finally ends with the robot development and control. This methodology could potentially be applied to develop other types of robots from certain animals that do not employ dynamic gaits.

The target of developing such a robot is two-fold: (1) to produce a physical tool for neuroscientists to evaluate their control algorithms and test their hypotheses in a platform that comes closer to the behavior of the real animal and (2) to produce a robotics tool to study quadrupedal sprawling 3D locomotion and tail function based on a more realistic model.

Another important goal of this project is to properly characterize the walking gait of salamanders. To date, studies on salamander kinematics have used conventional techniques to track the body or limb movements and they have focused either on the spine or on the limbs. The most commonly used techniques place markers on the skin. However, the movement of the skeleton is the one that most accurately describes the actual movement of the animal. These data are not yet available for the salamanders. In addition there is no comprehensive study to describe the kinematics of the forelimbs, scapula movements, limb rotations, and detailed kinematics of underwater stepping (although there are studies from [24], [4], and ([10]) associated with underwater stepping and forelimb kinematics, the detailed study is missing).

### 1.3.1 Summary of Report

The project report is outlined as follows: The kinematic data collection and acquisition will be discussed in chapter 2. Chapter 3 is dedicated to the analysis of the kinematics data collected in Chapter 2. Later in the report, the spine joints optimization and limb joint simplification to create an optimal model of the salamander joints will be discussed (chapter 4). The mechanical design, tips and criteria, and also robot development will be covered in chapter 5. Before testing the kinematic data extracted from the real salamander on the robot, the model will be simulated to verify the correctness of modeling as well as kinematic data of locomotion. The physical simulation will be presented in chapter 6. Robot hardware and control methods will be investigated in chapter 7. This chapter describes the hardware and electronics used in the robot and also the control technique and the kinematics applied to the robot. Chapter 8 is dedicated to the discussion of the project as a

### 1.3. PROJECT GOALS AND OBJECTIVES

whole. Finally the report will be concluded by chapter 9 in which the concluding remarks and future work will be presented.



## Chapter 2

# Kinematic Data Collection

In order to characterize different gaits of salamanders locomotion, it is necessary to use techniques that could reproduce the locomotion behavior of the animal. Such techniques aim to reflect the gaits parameters and characterize them so that they represent the animal locomotion. Salamanders in general, employ different gaits, the most important gaits that are subject of studies are terrestrial stepping, underwater stepping and swimming. To accurately study the kinematics of the locomotion, it seems rational to record the animal movements and later try to assess the behavior in detail. The fundamental objective is to extract the kinematics out of the recorded videos in one complete cycle of locomotion by the assumption that the locomotion is symmetric and repeatable. The cycle is defined as the time in which a particular event is repeated (e.g. limb footfall). In salamanders, this could for instance be the time when one limb touches the ground until the next footfall on the ground. In this chapter we will describe possible methods to collect the kinematic data and what method we used in detail.

### **2.1 Different Methods of kinematic data collection:**

Several techniques are applied to study the animals' kinematics. [3] obtained the kinematic data of the slider turtle in lateral and ventral views using two digitally synchronized high-speed video cameras in which a combination of white and black marker pen were used to draw high-contrast points on some anatomical landmarks (e.g. tip of the nose, shoulder, elbow, wrist,etc). The recorded videos are used to extract the kinematics. Similarly, [28] used some reflective landmarks painted on the lizards to mark the vertebral column and the pelvis, knee, ankle, etc. The animal was then videotaped in lateral and dorsal views. Moreover, [23] used marker points in some particular parts of the salamander. The dots were painted along the body (e.g. dorsal ends of the two ilia, at the left knee, at the distal end of the left fibula, and some equally spaced points beginning at a point between the two scapulae and ending between the two ilia). Then the animal was persuaded to walk on a treadmill, while it had been video-recorded. The system setup was equipped with

## CHAPTER 2. KINEMATIC DATA COLLECTION

two cameras; one for a lateral view and the other for the dorsal view. The videos later were analyzed to obtain the kinematics. [24] used anatomical landmarks on some body parts (e.g. vertebral column midway between the shoulders, elbow, wrist, hip, knee and ankle joints) where sequences of images from the dorsal and lateral views were digitized independently. [13] video-recorded the swimming salamander and used the videos to mark some points along the vertebra. In this work, X-Ray photograph of the salamander was used to obtain the size and shape of axial elements along the vertebral column.

In addition to the landmark video recording techniques explained above, some studies have focused on the kinematics through assessing the muscle activities along the kinematic markers. [22] used electro-myographic (EMG) recording of axial muscles to investigate the kinematics of the salamander in swimming and trotting. Furthermore, white markers were used at the skin of the animal to serve as kinematic reference points. [25] used painted dots in the body while at the time using EMG for hind-limb muscles activities. [26] studied the lizard locomotion by placing some markers and EMG electrodes in the trunk of the animal. Moreover, [8] investigated the patterns of activation of axial and limb muscles during both swimming and terrestrial locomotion by means of EMG recordings.

Another method that has been used is X-ray cinematography. [11] employed X-ray technology to analyze the kinematics of 8 species of mammals. The individuals were subjected to move on a motor-driven treadmill while the X-ray equipment recorded in a lateral projection. With similar approach, [9] used separately recorded x-ray perspectives to create 3-D model of the kinematic data in lizards. Furthermore, [12] introduced scientific rotoscoping (SR) method in which they simultaneously animated and quantified animal's skeletal movement by considering a 3D computer model according to x-ray and standard video data. In this method, first an experimental scene is reconstructed in 3D, and then the skeleton model is created from the CT or laser scans. They describe examples of pigeon and alligator walking.

Studies on salamander kinematics have used conventional techniques in order to track the body movements and they have focused either on the spine or the limbs. As discussed above, most of the techniques place markers on the body to track the locomotion of the animal. It seems as though that the movements of the skeleton is the one that most accurately describes the actual movement patterns of the animal and could potentially represent the closest kinematic estimation of the real animal. As a subject matter of the kinematic study, this method has not been applied until now for salamanders.

In this project recorded X-ray videos are used to investigate the locomotion of the salamander in different gaits such as ground stepping, underwater stepping and swimming. A set of recorded X-ray videos both in lateral and dorsal views are used to obtain the kinematic data.

## 2.2. DATA COLLECTION FROM X-RAY VIDEOS

### 2.2 Data Collection from X-ray Videos

We have a set of X-ray videos in dorsal and lateral views. These views could be applied to have the data in 3D Cartesian space; the dorsal view could be used for the X-Y coordinate while the lateral view for the Z-coordinate. To obtain the kinematic data out of the recorded videos a software called Mtracking was developed in BioRobotic Laboratory, EPFL<sup>1</sup>. This software enables the user to manually put the points of interests in a given X-ray video. Using the X-ray technology, we could see the skeleton of the animal and track the desired bones and joints.

After loading, the frames of each video could be sampled with arbitrary number (e.g. sample every 20 frames). In each sampled frame of the video, the user could track some points which are the important joints involved in the animal locomotion. The process of tracking will be repeated frame by frame until we reach a frame that corresponds to a complete cycle of the locomotion. In walking, the cycle has been defined as the hindlimb (left or right) contact with the ground to the subsequent contact. The output of Mtracking is a text file with a set of coordinates in pixels which corresponds to the tracked points. These data files later were imported in MATLAB to convert to desired types.

Table 2.1 summarizes the important tracked points for each video. As inferred from the table, the scapula bones are only visible in dorsal view and not in lateral view. Thus we only tracked them in the dorsal view and for the lateral view we gave the same coordinate as the shoulders for the corresponding points. Since the vertebra in the tail are not visible as they are in the trunk, we track some arbitrary number of points along the tail. The methodology of extracting the tail points will be discussed later.

**Table 2.1.** The tracked points in the skeleton of the salamander. There are 3 points in the head, 2 points for each scapula bone (only dorsal), 5 points in each limb (forelimb and hindlimb), 16 points along the vertebral column in dorsal view and 9 points in lateral view, and arbitrary points along the tail.

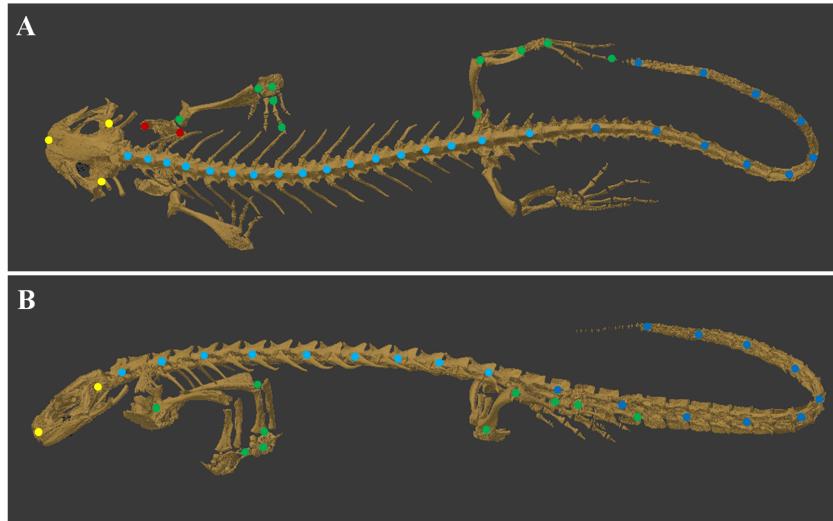
Body Part	Tracked Points Dorsal/Lateral
Head	Snout (Tip of nose) Left Jaw Right Jaw
Fore-Limb	Scapula Back (only in dorsal view) Scapula Front (only in dorsal view) Shoulder Elbow Wrist Middle Finger Root Middle Finger Tip
Hind-Limb	Hip Knee Ankle Middle Finger Root Middle Finger Tip
Trunk	Dorsal view: 16 points along the vertebral column Lateral view: 9 points along the vertebral column
Tail	Arbitrary points along the tail

---

<sup>1</sup>by Konstantinos Karakasiliotis

## CHAPTER 2. KINEMATIC DATA COLLECTION

Fig.5.1 which is the Computed Tomography Scan<sup>2</sup> of the salamander skeleton, schematically illustrates the tracked points. The colored dots in both figures are the points summarized in Table 2.1.



**Figure 2.1.** The Computed Tomography Scan (CT Scan) of the salamander skeleton. (A) Dorsal view, 3 points on the head (yellow), 2 points on each scapula (red), 5 points in each limb (forelimb and hindlimb shown in green), and 16 points in the spine along the vertebral column (light blue), number of tail points are arbitrary (dark blue) (B) Lateral view, 3 points in the head where the two jaw points are superimposed (yellow), 5 points in each limb (forelimb and hindlimb in green) and 9 points in the spine (light blue), number of tail points are arbitrary (dark blue).

Fig.2.2 (A) and (B) shows one frame of a ground walking video in Mtracking in dorsal and lateral views respectively. As discussed previously, we tracked 16 points along the vertebral column in the trunk in dorsal view and 9 points in the lateral view. Since we only use the lateral view to infer the height (Z-axis), we later fit a spline to these 9 points and use it to obtain the height of 16 tracked points in the dorsal view.

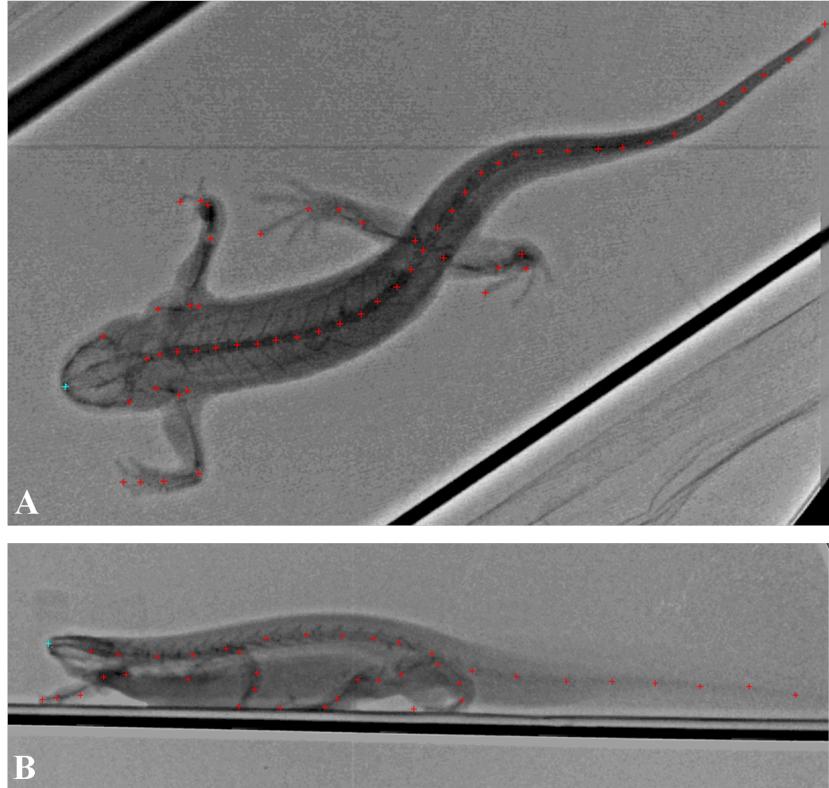
As mentioned already, each video has a side (lateral) view and a top (dorsal) view. The dorsal view is used to get the X-Y coordinates of each joint while the lateral view is used to infer the height (Z coordinate) of the corresponding points in the dorsal view. We tracked the X-ray videos from ground walking, underwater walking and swimming<sup>3</sup>. These data will be later analyzed accurately to obtain the desired kinematic information.

---

<sup>2</sup>CT Scan by Alexander Haas from the university of Hanover and Nadja Schilling.

<sup>3</sup>The ground walking X-ray videos were tracked by Konstantinos Karakasiotis, underwater stepping by Navid Khajeh Mahabadi and swimming by Jonathan Grizou

### 2.3. DATA CONVERSION AND CALIBRATION



**Figure 2.2.** One frame of ground stepping, X-Ray tracking of the real salamander.  
(A) Dorsal view, (B) Lateral view.

## 2.3 Data Conversion and Calibration

The objective of data conversion is to read the tracked points stored in the text files and convert them to the Cartesian space. The tracked points stored in text files, represent the pixel coordinates of the points in the X-ray videos. We used MATLAB to convert this information to Cartesian space in 3D so that the data would be ready to analyze and obtain any necessary information.

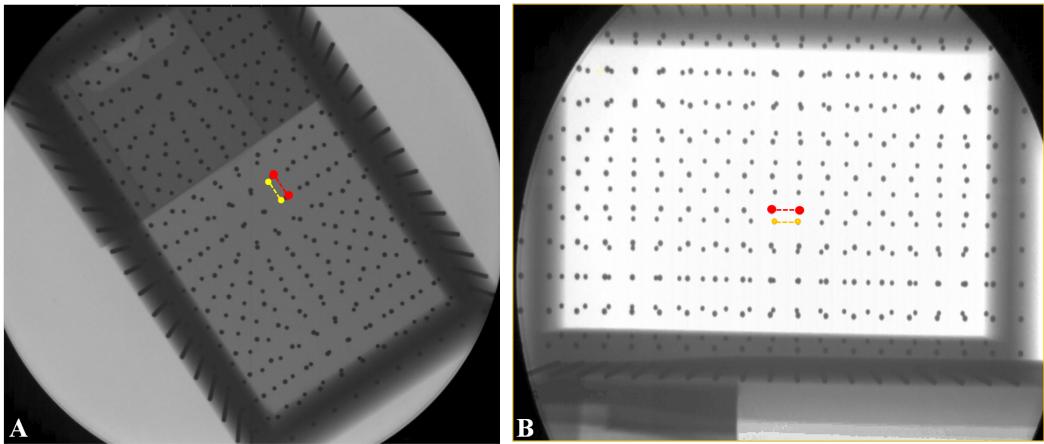
### 2.3.1 Calibration Methodology

One important step of data reading, is to calibrate and scale the recorded videos. This means the translation from video coordinates (in pixels) into Cartesian coordinates (in centimeter) while trying to make the output dimension as much close as to the real animal. This stage includes the conversion from pixel to centimeter and also compensation of the camera distortion and the perspective effects. For the camera distortion correction, a set of lookup tables were provided<sup>4</sup> to compensate for the distortion. For the sake of calibration and scaling two calibration images

---

<sup>4</sup>"By Konstantinos Karakasiliotis"

were provided. These images were obtained from a "calibration cube" that was pictured in the X-ray setup. The images showed the top and side views of the cube. The cube structure was such that there were equally spaced small dots in the surrounding surfaces and the distance between each pairs of adjacent dots was 1cm (Fig.2.3 (A) and (B)). In Fig.2.3 (A), the smaller dots (shown schematically in yellow) correspond to the lower plane while bigger dots (shown in red) correspond to the upper plane of the cube.



**Figure 2.3.** Calibration cube; (A) Dorsal view, the red and yellow points correspond to the upper and lower planes of the cube respectively, the red and yellow lines correspond to  $U_d$  and  $L_d$  in dorsal calculations respectively (B) Lateral view, the red and yellow points correspond to the front and back planes of the cube respectively, the red and yellow lines correspond to  $U_d$  and  $L_d$  in lateral calculations respectively.

Now consider the calibration in dorsal view. In Fig. 2.3 (A), denote the distance (in pixels) between each adjacent pair of the the upper plane (red dots) with  $U_d$  and the one for the lower plane (yellow dots) with  $L_d$ . We know that the distance (in metric system) which corresponds to  $U_d$  and  $L_d$  is 1cm. We also know that the salamander was placed somewhere in between these two planes. The perspective effect is proportional to the position that the salamander is placed with respect to these planes. This means that if for example the salamander is placed closer to the lower plane, the resulting image would be smaller compare to the case when the salamander is placed closer to the upper plane. This situation is similar to an object that is placed below the light source and its shadow is on the ground, the closer the object to the ground the smaller the shadow.

Therefore, if we observe the tracked points of the salamander from the dorsal view, the depth that corresponds to each point (this is obtained from lateral view) is important to calculate the pixel to centimeter conversion ratio, i.e. for each point (in dorsal view) we have a different ratio since we have a different corresponding depth (in lateral view). Note that the depth that corresponds to the point in dorsal view is obtained from the lateral view and is different for each point. Accordingly, for each given point in the dorsal view, the pixel to centimeter ratio is calculated

### 2.3. DATA CONVERSION AND CALIBRATION

as:

$$pix2cm = (U_d - L_d)z + L_d \quad (2.1)$$

where  $z$  is the depth that corresponds to a given point in dorsal view. This depth is calculated in percentage as the relative distance (in pixels) with respect to the upper and lower planes . Note that Eq. 2.1 should be calculated for each point in the dorsal view due to the fact that each point has a different depth. This formula is derived to compensate for the perspective effect of the camera in lateral view and to calculate the conversion ratio.

Similarly, this conversion could be applied for the lateral view. With the same scenario we could easily calculate the pixel to centimeter conversion ratio in the lateral view. In this case the depth is obtained from the dorsal view. With the similar definitions,  $U_d$  and  $L_d$  would be calculated from Fig. 2.3 (B). Using Eq. 2.1, the conversion ratio from pixel to centimeter would be obtained for each point in the lateral view. Here, "z" is the depth that corresponds to a given point in lateral view and it is calculate as the relative distance with respect to front and back planes of the cube.

Having obtained the pixel to centimeter conversions, the next step is to rotate the entire points in dorsal and lateral views to make them parallel to the horizontal axis (see Fig. 2.2, (A), (B)).

#### 2.3.2 Conversion Tips

The tail was tracked arbitrarily since the vertebrae in the tail were not visible. We put the points on the tail so that they fit to the tail as a curve and could represent the tail. The trick in reading and conversion was that we fit a spline to the tracked point of the tail (in dorsal view), and then we separated some equally spaced segments from the fitted curve. The distance of each segment was equal to one vertebra which is approximately 0.35cm. The fitted curve was smoothed by MATLAB cubic smoothing spline in order to reduce the tracking errors and have a smoother data.

As discussed previously, the lateral view was used to infer the height (Z coordinate) of the points and the dorsal view for the X-Y coordinates. For each tracked point in dorsal view, we have a corresponding height which is tracked in the lateral view. For example, the X-Y coordinates of tip of nose is obtained from the dorsal view and this point has a corresponding height (Z coordinate) which is obtained from the lateral view.

However, for the points in the spine, since we tracked the points in dorsal and lateral views differently, for each point in dorsal view we do not have a corresponding point in lateral view. Therefore, in order to determine the height of the tracked points in dorsal view, we fitted a spline to the spine points in the lateral view. By interpolating the fitted curve which contains the height information of the spine points we could easily obtain the height for each point. The fitted curve was also smoothed to reduce the tracking errors.

## CHAPTER 2. KINEMATIC DATA COLLECTION

In conclusion, we read the tracked data that were stored in text files and converted them into Cartesian coordinates in 3D for each X-ray video. This information was stored in MATLAB mat files for each video. From now to the end of the project we will use these saved data to obtain necessary information.

## Chapter 3

# Kinematic Data Analysis

In order to characterize the locomotion gaits and study the kinematics of the salamander it is important to analyze the X-ray videos obtained from the tracking. These data are not only used to study the kinematics but also to extract the necessary information for modeling, optimization, design and development of the robot during the entire project which will be discussed in the next chapters. As we pointed out in the previous chapter, we tracked different gaits of the salamander in terrestrial and aquatic locomotion (ground stepping, underwater stepping and swimming) and we used MATLAB to read the tracked videos and convert them to Cartesian space. However in this chapter we only focus on the analysis of the ground stepping gaits. The deliverable of the previous chapter was the tracked points in 3D Cartesian space and we tracked different parts of the salamander such as limbs, spine, head, and scapula. In this chapter we first discuss the software development that aims to facilitate the analysis and in general to obtain required information. Then we present the results of the kinematic analysis.

### 3.1 Software Development for Data Analysis

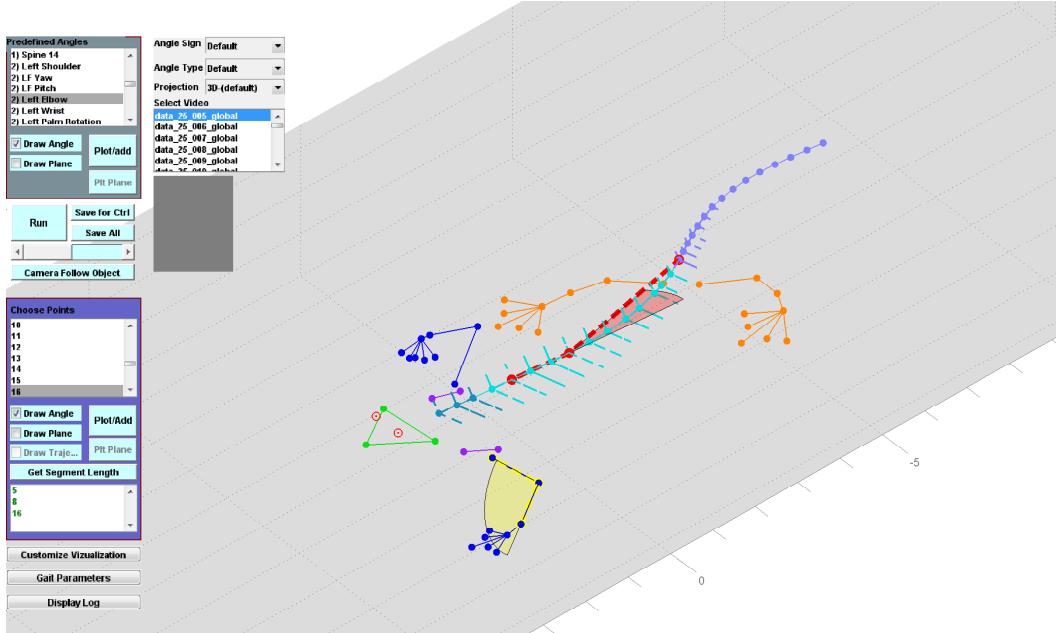
In order to facilitate the analysis, a graphical user interface (GUI) was developed in MATLAB. The interface enables the user to select the saved videos<sup>1</sup>, and visualize the tracked points so that they represent the skeleton of the salamander. Fig. 3.1 illustrates how the GUI looks like and the tracked points that reflect the skeleton. In this figure you can see the points in 3D and there are a lot of options to work with this data. We only track one finger per limb, the other fingers shown in the figure are just visualization. This is also true for the ribs and the eyes.

One feature that makes the interface so useful is that there is a list of pre-defined angles which the user could select and plot the angle variations during the tracked cycle of locomotion for instance. The definitions of these angles will be discussed later in this chapter. There are some control features developed in the interface that make it possible to visualize the selected angle. This way the user could see the

---

<sup>1</sup>The deliverables of previous chapter; the X-ray videos from the tracking

### CHAPTER 3. KINEMATIC DATA ANALYSIS



**Figure 3.1.** The Graphical User Interface developed mainly for the analysis and also other data extraction, the yellow angle is the left elbow angle selected from the pre-defined list, and the red angle is defined by the user on the spine. As shown in the left side of the figure, there are many options that the could select. The user could see and plot different angles (pre-defined/arbitrary) or their projection in different plane, it is also possible to save the selected angles/videos.

selected angle visually and better understand the angle definition. Fig. 3.1 shows a yellow arc which is the angle of the left elbow as an example. Furthermore, every angle defines a plane which could be easily drawn by the user to see the variations of the angle plane. Also these variations can be plotted with respect to different coordinate planes (e.g. X-Y, Y-Z, etc).

In addition to the list pre-defined angle, there is a list of points which are in fact the tracked points (e.g. head, spine points, limbs, ect). This feature make it possible for the user to define an arbitrary angle. All the graphical features mentioned already, also applies here. Fig. 3.1 illustrates a red angle defined in the spine of the salamander for instance. The user can also calculate the length of the selected sequence of the points.

There are also some other control features for angle calculations such as projection in different planes, supplementary angles definitions (acute, obtuse, etc), sign definitions, etc. There is a run button in the interface that enables the user to play all tracked frames of the video as a sequence so that the entire frames seems like a video. Also a slider in the interface makes it possible to see the frames step by step. This is useful for example to visually observe the angle variations in different frames. Moreover the user is able to select several videos/angles to save them as a database for further analysis. Another important feature of the interface is to cal-

### 3.2. DEFINITION OF VARIABLES

culate and save the angles as lookup tables for robot control and simulation which will be discussed in later in this project. In summary, the interface provides the opportunity to calculate different angles, save databases, analyze different gaits and in general to extract any required information.

## 3.2 Definition of Variables

In stepping, a "cycle" is defined as a hindlimb (left or right) contact with the ground to the subsequent contact of the same limb. In the analysis, the right hindlimb is considered as the reference for all of the analysis. During the cycle, the time that the limb (fingers followed by foot/palm) is in contact with the ground is called the "stance phase" and when it loses contact with the ground (e.g. to move forward) is termed as the "swing phase".

The following angles were defined in 3D for the analysis:

- **Spine angle:** having 3 points that forms the angle, spine angle is defined as the angle between the lines connecting pairs of adjacent spine points. There are 2 possible angles with this definition, we choose the angle that is shown schematically in Fig. 3.2 (A). As it is apparent from the figure, zero angle happens when the two lines are along each other. Starting from the zero angle and taking the first line (the one that is closest to the head) as the reference, if the second line rotates counter-clockwise the created angle would be positive otherwise it is negative. The figure hence shows a positive angle.
- **Forelimb angles:**
  - The shoulder joint employs three degrees of freedom; "roll", "yaw" and "pitch". Depending on the order of these three angles, we could have different values. We selected the order "roll, yaw and pitch" for the analysis since this sequence seems more intuitive. Putting a local right-handed frame on the shoulder in which the x-axis is parallel to the reference in the spine and the limb lies on the y-axis when all angles are zero, the roll, yaw pitch are defined as rotation around y, z and x axes respectively (Fig. 3.2 (B)). Note that frame is put on the shoulder such that the local x-y plane of the frame is always parallel to the ground. The right hand rule is applied to determine the sign of each angle.
  - **Elbow angle:** the angle between the line connecting the shoulder to elbow and the line of elbow to wrist (as shown in Fig. 3.2 (C), green angle). Taking the shoulder-elbow line as the reference, zero angle happens when the line of elbow-wrist and the reference superimposed.  $180^\circ$  occurs when elbow-wrist is straight with the reference. This angle is always defined to be positive.

- **Wrist angle:** is quite similar to the elbow angle in terms of definition, and is the angle defined with the lines of elbow-wrist and wrists-finger root (see Fig. 3.2 (D), green angle).

- **Hindlimb angles:**

- Similar to shoulder, the hip also contains three degrees of freedom; "roll", "yaw" and "pitch". Taking the sequence of "roll, yaw and pitch", the angles are defined as the rotation around the y, z and x axes respectively. We put a local frame on the hip. To define the local frame, we took two points of the spine (before and after the hind girdle) as the reference (Fig. 3.2 (B), red line). The local x-axis is parallel to this reference and the limb lies along the y-axis when all angles are zero. The x-y plane of the frame is parallel to the ground and angles sign obey the right hand rule.
- **Knee angle:** is defined as the angle between the lines of hip-knee and knee-ankle. All other definitions match that of the "elbow angle" (Fig. 3.2 (C), red angle).
- **Ankle angle:** is the the angle of knee-ankle with the knee-finger root which is similar to the "wrist angle" (Fig. 3.2 (D), red angle).

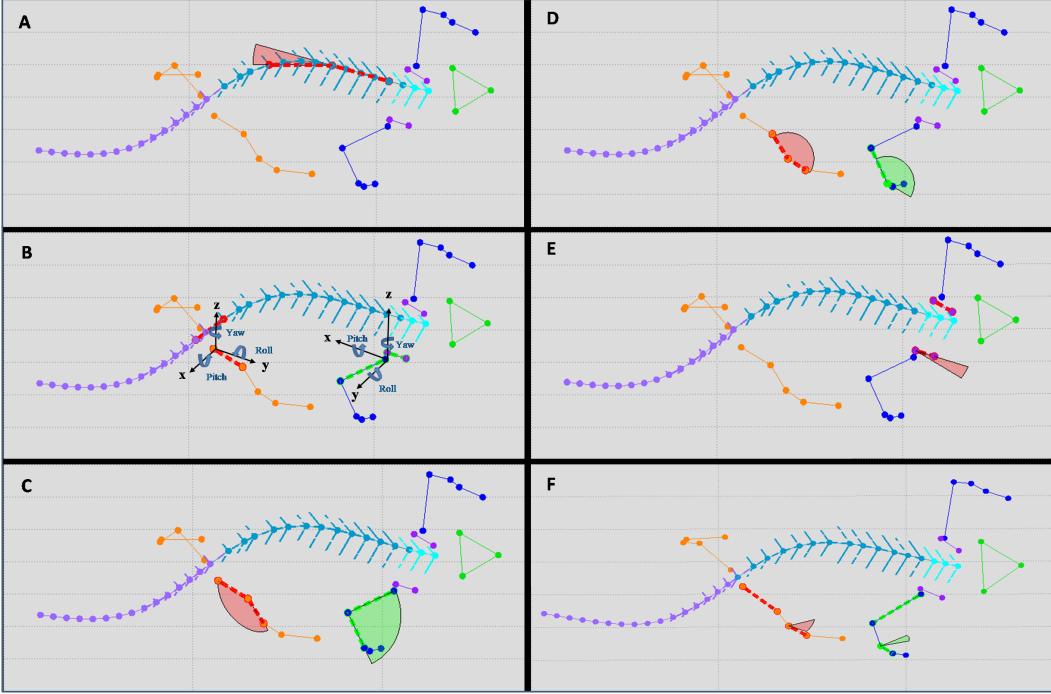
- **Special angles:**

- **Scapula angle:** is defined as the angle that the lines of scapula bones create with each other as illustrated in Fig. 3.2 (E), the example shows a negative angle.
- **Foot rotation:** In the hindlimb this angle is supposed to describe how much the foot will rotate during the locomotion (specially in stance phase). Consider a plane with normal vector of the tibia (knee-ankle line). Let's call this plane the "tibia" plane. The angle is defined as the angle between the projection of the "ankle-finger root" line on tibia plane and the projection of the "hip-knee" line on the tibia plane. Fig. 3.2 (F), indicates this angle in red.
- **Palm rotation:** This angle is similar to the foot rotation angle. Consider a plane with normal vector of the ilia (elbow-wrist line). Let's call this plane the "ilia" plane. The angle is defined as the angle between the projection of the "wrist-finger root" line on ilia plane and the projection of the "shoulder-elbow" line on the ilia plane. The angle is shown in Fig. 3.2 (F) with green color.

### 3.3 Results

This section addresses the results of the kinematic analysis in the walking gait of the salamander. The analysis of the spine and limbs angles will be discussed briefly.

### 3.3. RESULTS



**Figure 3.2.** Representation of angle definitions for the analysis. (A) A typical angle on the spine, this angle is positive in sign, (B) Roll, yaw and pitch defined for the shoulder/hip, the local frame is put on the shoulder/hip such that the x-axis is parallel to the reference scapula/spine line and the x-y plane is parallel to the ground, (C) The angles of elbow/knee, these angles are always positive with zero when the two parts of the limbs forming the angles superimposed, (D) The angles of wrist/ankle, these angles are always positive with zero when the two parts of the limbs forming the angles superimposed, (E) The scapula angle which is the angle between the lines of scapula bones, (F) The foot/palm rotation angles. This angle indicates how much the foot/palm rotates during the locomotion.

Two individuals have been taken into account in the analysis. In total, the two individuals employ 23 cycles of locomotion.

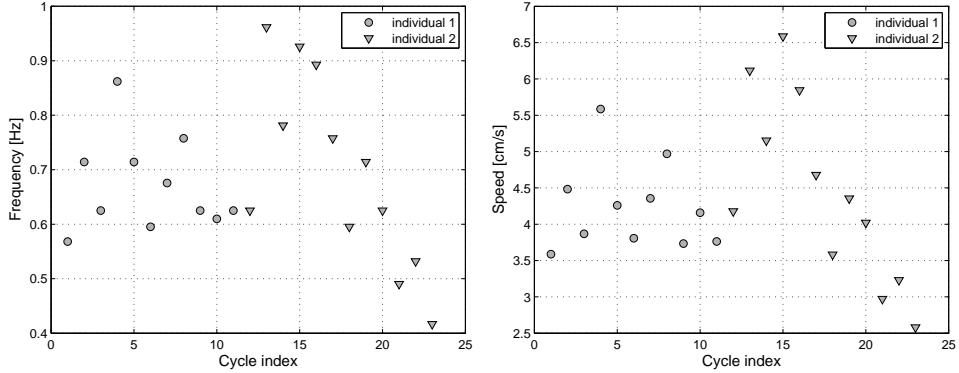
One feature that is interesting in the analysis is the frequency and the speed that the salamander use during walking. Fig. 3.3<sup>2</sup> illustrates the frequency and the speed ranges that the individuals employ during the walking gaits. Fig. 3.3, left shows the frequencies (Hz) in the vertical axis, and Fig. 3.3, right indicates the speeds (cm/s). As the figure implies, 11 cycles are employed by the first individual and 12 cycles by the second.

To characterize the walking gait of the salamander it is important to know the time when the foots are on the ground (stance phase) and when they are not (swing phase). Therefore, the footfall diagram of walking gait was determined which is depicted in Fig. 3.4. The solid bars indicate the periods which the foot is on

---

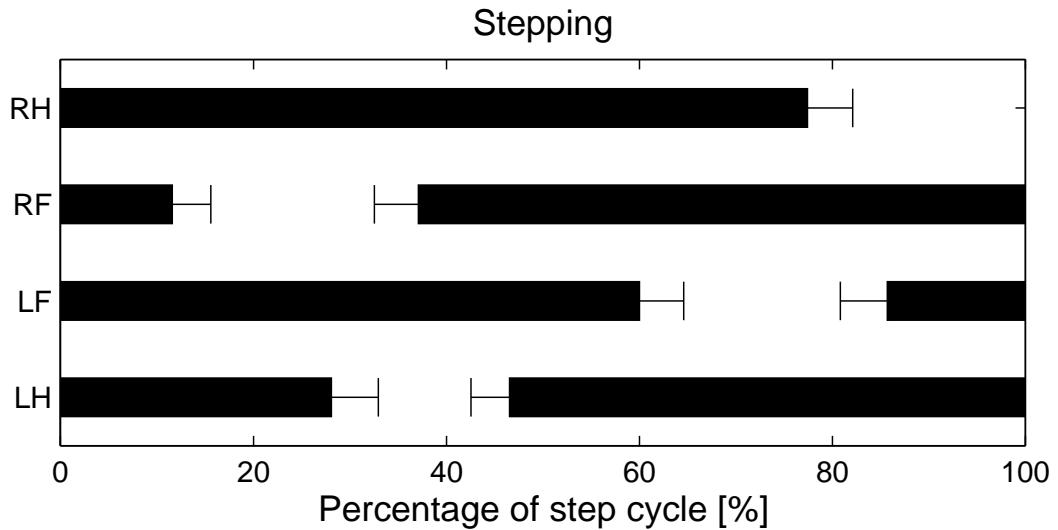
<sup>2</sup>The figures of this section is obtained by "Konstantinos Karakasiliotis"

### CHAPTER 3. KINEMATIC DATA ANALYSIS



**Figure 3.3.** Left: Frequency ranges that the two individuals exhibit during walking gaits, Right: the speed ranges of the individuals during the walking gaits. 11 cycles are employed by the first individual and 12 cycles by the second individual.

the ground and the end bars are the standard deviations. The data represent the results for all 23 cycles of the two individuals. The vertical axis is the limbs (Right Hindlimb, Right Forelimb, Left Forelimb, and Left Hindlimb) and the horizontal axis is the percentage of the cycle.



**Figure 3.4.** The footfall diagram of walking, solid bars indicate the periods which the foot is on the ground and the end bars are the standard deviations. The vertical axis represents the foots; RH: Right Hindlimb, RF: Right Forelimb, LF: Left Forelimb, and LH: Left Hindlimb. The horizontal axis is the percentage of the cycle.

The statistical analysis of the angles on the spine is illustrated in Fig. 3.5. In the analysis we have considered all cycles for the two individuals. The angles from the neck to the tail are depicted in the figure. The numbers in the left column of

### 3.4. DISCUSSION

the figure show max/min values of each angle and the numbers to the right column indicate the values of the phase lags of each angle. The circles show the mean value of the zero crossings. By zero crossing we mean when the angle crosses the horizontal axis. Also, the deviation of the zero crossings are illustrated by horizontal bars. The lines connecting the circles show the standing wave the trunk and a part of tail. The deviations of each angle is depicted in gray area and the mean value of each angle is shown by solid curves.

Fig. 3.6 indicates the results of the limbs angle analysis for the all cycles of the two individuals. The left figure shows the hindlimb and the right one shows the contralateral forelimb. The footfall diagram is also attached to the top of each figure to have an idea of when the footfalls are happening. The vertical axes are the angles ( $^{\circ}$ ) and the horizontal ones are the percentage of the cycle. For each angle, the deviations are shown in gray area and the mean value in solid curves.

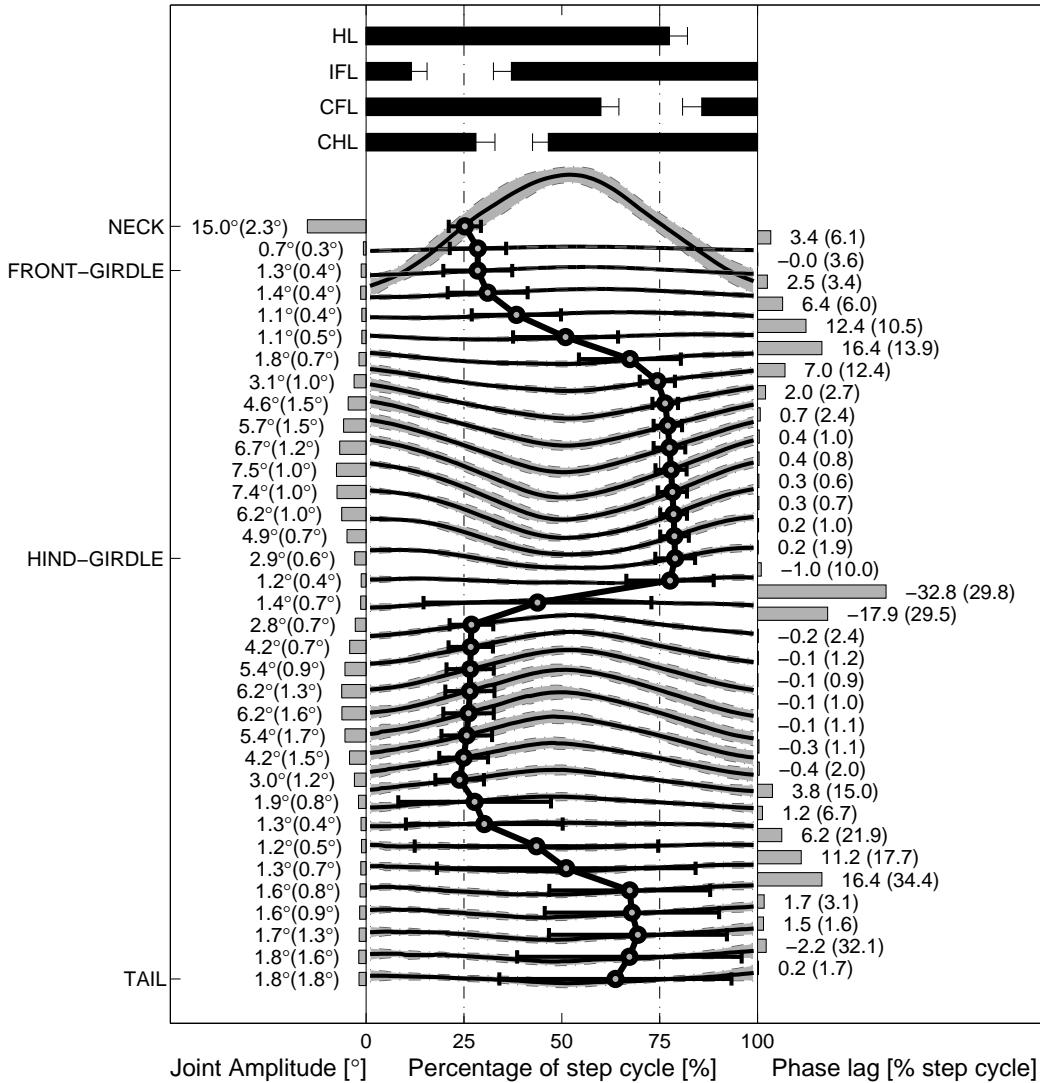
## 3.4 Discussion

According to the Fig. 3.3, the frequency of walking is between 0.4  $Hz$  and 0.95  $Hz$ . In addition the speed ranges between 2.5  $cm/s$  and 6.6  $cm/s$ . Although the environment and the conditions of the taping might have some effect of the frequency and the speed of the locomotion, these values are something that were expected in the walking gait.

The footfall diagram of the walking gait (Fig. 3.4) implies the time when the limbs are on the ground during the cycle. The lateral sequence walk is thought to be stable since the center of mass is always within the triangle formed by three limbs. This is indeed necessary for a animal with sprawling posture, however, according to the figure, a small portion of cycle is dedicated to the support by three legs. Instead a larger portion of the cycle is dedicated to the support by the diagonal limbs.

As expected, the spine employs a standing wave in the walking gait. As Fig. 3.5 implies, the lines connecting the circles show the standing wave pattern in the trunk (between the nodes close the girdles) and also the first portion of the tail. The largest angle in the spine, as shown in the figure, is the angle of the neck. Except for this angle, the largest amplitudes of the angles happen in the middle of the trunk and the smallest close to the girdles. When the angles in the middle trunk are at maximum, the middle angles in first portion of the tail are also at maximum (absolute value) but in the negative phase lag. This explains the S-shape in the spine during the walking gait.

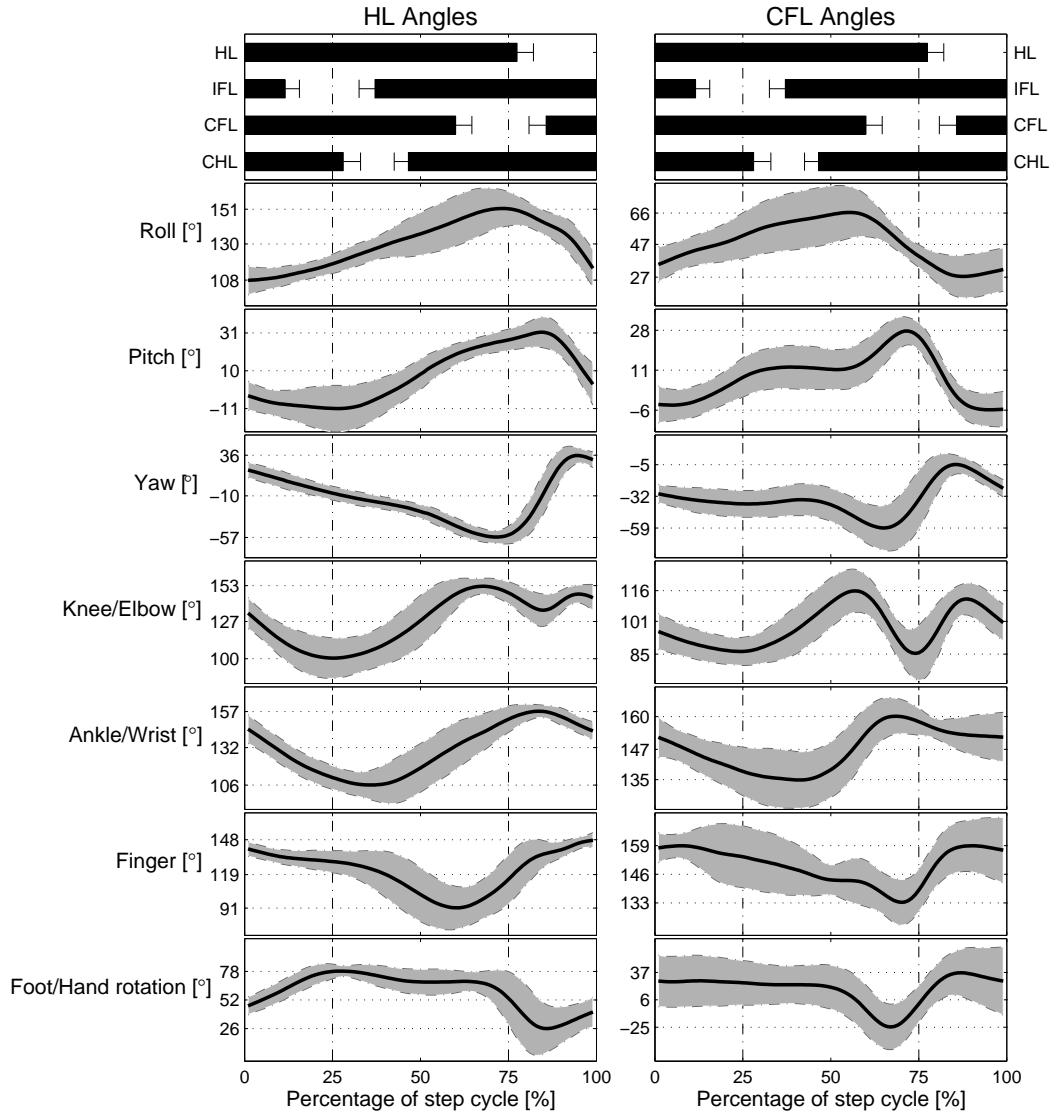
The results of the limbs analysis are mentioned in Fig. 3.6. The footfall diagrams, depicted at the top of the figures, are helpful to observe different angles in the swing and stance phases of the limb. As the figure implies, the limbs (forelimb, hindlimb) employ rolling during the locomotion, the variation of rolling is intense in the swing phase. The rolling here could be interpreted as the rotation of the entire limb. The pitch and the yaw angles here act together to protract (retract) and lift up (lift down) in the stance and swing phases. The rotation of the foot is



**Figure 3.5.** The statistical analysis of the spine angles from the neck to the tail. The footfall diagram is shown at the top of the figure; HL: hindlimb, IFL: ipsilateral forelimb, CFL: contralateral forelimb, CHL: contralateral hindlimb. The numbers in the left column of the figure shows max/min values of each angle (min is shown in the parenthesis). The numbers in the right column of the picture shows the min/max values of the phase lags with respect to the percentage of the cycle with max in the parenthesis. The gray area shows the variation of the particular angle, and the solid curve is the mean value. The circles show mean values of the zero crossings of the angles with the horizontal axis. The horizontal bars show the deviations of the circles. The horizontal axis is the percentage of the cycle.

remarkable in that it supports the limb during the stance phase.

### 3.4. DISCUSSION



**Figure 3.6.** The statistical analysis of the angles in the limbs. The footfall diagram is shown at the top of the figure; HL: hindlimb, IFL: ipsilateral forelimb, CFL: contralateral forelimb, CHL: contralateral hindlimb. The deviations of each angle are shown in gray area and the mean value in solid curves. The horizontal axes are the percentage of the cycle and the vertical axes are the angles ( $^{\circ}$ ).



## Chapter 4

# Model Optimization and Simplification

Similar to every modeling, the salamander model would be a simplified version of the complex structure of the animal which should describe and represent the kinematics data. Indisputably one important step in modeling is to optimize the number of joints while respecting the kinematics derived from the animal so that the model could reflect the animal gaits. It seems irrational and impractical to include all of the degrees of freedom of the animal in a complex model, on the other hand, some joints of the animal employs a complicated structure (e.g. in terms of muscle placement, activations, etc) which are in practice impossible to implement. Limitations such as motor physical properties, power consumption, control problem, mechanical complexity and integration results in considering a n optimal model for the robot. Besides a complex model with maximum number of joints does not necessarily possess much advantage compared to an optimal one. Thus a simpler model should be obtained for the kinematics representation of the animal with the optimal number of degrees of freedom. This is specially the case for the spine model.

There exists 16 joints in the trunk of the animal, each of which contains 3 degrees of freedom (i.e. roll, pitch and yaw). Also the tail contains lots of degrees of freedom. As discussed above, due to the mechanical constraints and other limitations it is not feasible to develop the model which contains all degrees of freedom. Thus the optimization plays a key role in modeling to achieve the optimal number and positions of joints that could represent an accurate kinematics of the animal.

According to the data obtained from the tracking, there are 24 tracked joints in the limbs (forelimb and hindlimb including the scapulas) of the animal. Because of the mechanical problems such integration, mass distribution and center of mass, we should consider a way to simplify the model of the limbs specially for the hindlimbs. The animal anatomy shows that some joints such as knees and elbows only employ one rigid degree of freedom while some other joints such as shoulders and hips are spherical joints which employ more than one degree of freedom. Moreover, the ankles and the wrists have an articulated structure which make it difficult to accurately analyze and replicate the behavior. Therefore these joints are not considered in the optimization and simplification problem. The spine joint optimization prob-

lem could be decoupled from the hindlimb simplification problem since the effect of spine on the locomotion could be studied separately from that of the limbs.

In this chapter we will discuss the optimization problem for the spine of the animal and also the hindlimb model simplification. The objective is to find the optimal number of the joints and also calculate the best positions of these joints. The method and the results of this optimization will also be addressed in this chapter. Moreover, the hindlimb kinematic simplification will be described at the end of this chapter.

## 4.1 Spine Joint Optimization

The optimization problem of the tracked joints in spine is twofold. First we should find the number of optimal joints that can replace the actual number of tracked joints. Second, the positions of these optimal joints should be found. The problem can be solved mathematically, while the spine is considered as a set of curves. In different gaits of locomotion (walking, swimming, etc) the spine is a variable curve with different bending and curvature and the solution should fit into all of the gaits.

In order to simplify the problem, the spine model is considered in 2D instead of 3D. In other words, we consider that the spine model is planar and each joint employs one degree of freedom. Although in the animal, the spine joints are acting in 3D and they exhibit pitching (i.e. rotation around the Y-axis<sup>1</sup>), yawing (rotation around the Z-axis), and also rolling (rotation around the X-axis). However the most significant degree of freedom is yawing and we therefore consider a 2D spine model that only uses yawing.

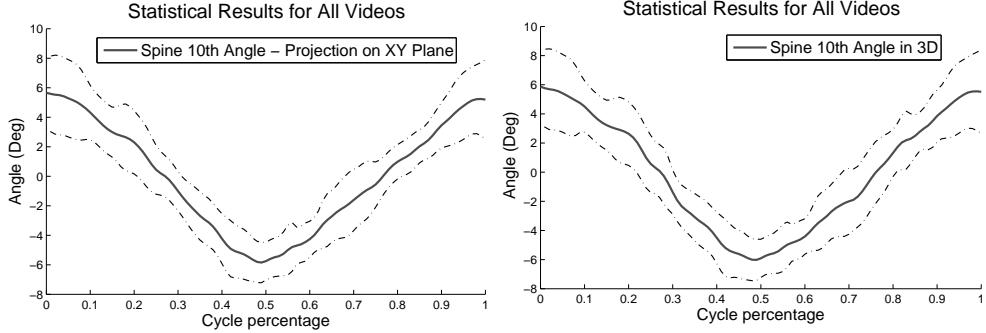
The reason that we can consider a 2D model lies on the fact that the spine angles are quite similar in 3D and 2D. Fig 4.1 illustrates the angle of a typical spine joint (e.g. 10<sup>th</sup> joint) in the 3D space and its projection on the X-Y plane (2D) in one cycle of the locomotion. This angle is calculated statistically for all videos in order to verify this simplification. The dashed lines are the variations of the angle and the solid line is the mean value. The figure indicates that there is not a remarkable difference between angles in 3D and 2D which further demonstrates that the simplification is valid. Note that for the other spine angles, we get the same results and thus we can conclude that considering the spine model in 2D is valid.

As mentioned already, the optimization problem is to find two items; number and also the position of the joints. The optimization problem can be further simplified by fixing one item and try to find another item. It is more convenient to fix the number of joints and focus on finding the optimal positions. This means that one can start with an arbitrary number of joints (e.g. 3 joints) for the spine and try to find the best positions of them on the spine. Assume that we have found a way to solve the optimization problem in terms of the positions. The next step is to

---

<sup>1</sup>In a right-handed coordinate system which is placed locally on a spine joint, X-axis lies along the spine, and Z-axis pointing upwards

#### 4.1. SPINE JOINT OPTIMIZATION



**Figure 4.1.** A typical spine joint angle statistically shown in one cycle of locomotion for all videos; Left: Angle in 2D, Right: Angle projection in 3D. The dashed line shows the variations of the angle and the solid line indicates the mean value of the angle for all videos.

increase this arbitrary number up to a reasonable number (e.g. from 3 up to 15). Increasing the number of joints will lead to a better estimation of the spine as a variable curve. The error comparison of different number of joints can give us an intuitive criterion to select the optimal number of joints. The error here can be defined as the area difference between the actual curve (spine), and the estimated one. Obviously the more the number of the joints, the better the curve estimation, however, a large number of joints is not optimal. Thus we should find a threshold to select the optimal number. When the optimal number of joints is obtained, the optimization problem is already solved since the best positions of these joints are known in advance.

We decoupled the optimization problem into position optimization and number optimization. In position optimization we try to find the optimal positions of a given number of joints in a curve. We can now discuss the possible approaches to solve this position optimization. The problem can be solved whether analytically or numerically. Although the analytical solution seems to be quite accurate, we will discuss in the following section, that it is in general difficult to solve the derived system of equations. Alternatively, the numerical approach tends to be more feasible yet without loss of accuracy. In the following sections we will discuss the possible approaches and then we will present the results of the spine position optimization. Later we will address the number optimization problem and we will conclude the optimization problem.

##### 4.1.1 Spine Optimization - Analytical Approach

Given a distinct curve, the problem here is to approximate the curve by a series of broken line segments which best fits the curve. The objective hence is to determine some intervals of the line segments so called "polyline" which result in the best curve approximation.

## CHAPTER 4. MODEL OPTIMIZATION AND SIMPLIFICATION

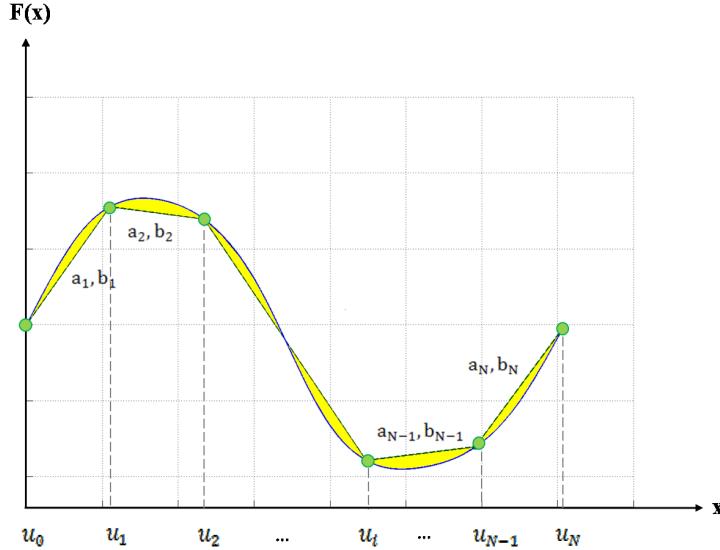
suppose that we have a curve which is described as follows:

$$y = f(x), \quad u_0 \leq x \leq u_N. \quad (4.1)$$

It is desired to break the given range, to the form:

$$y = \begin{cases} a_1x + b_1 & u_0 \leq x \leq u_1 \\ a_2x + b_2 & u_1 \leq x \leq u_2 \\ \dots & \dots \\ a_Nx + b_N & u_{N-1} \leq x \leq u_N \end{cases} \quad (4.2)$$

where in Eq. 4.2,  $a_i x + b_i$ ,  $1 \leq i \leq N$  are the equations of the line segments. Fig.4.2 shows the curve and the ranges. The optimal positions of the points on the curve occur when the yellow area is minimized.



**Figure 4.2.** An arbitrary curve, the objective is to find the best positions of the points that would best fit to the curve. To reach this the yellow area should be minimized.

In the least square sense, it is desired to find the unknowns  $a_i, b_i$ ,  $1 \leq i \leq N$  and  $u_i$ ,  $1 \leq i \leq N - 1$  that would minimize the yellow area. Going towards the solution, it is required to form a cost function which its minimization results in the optimal variables. This function can be described as the area between the curve and the approximated polylines. The following function can be formulated which is the interpretation of the area:

$$F(a_1, \dots, a_N, b_1, \dots, b_N, u_1, \dots, u_{N-1}) = \sum_{j=1}^N \int_{u_{i-1}}^{u_i} (f(x) - (a_i x + b_i))^2 dx. \quad (4.3)$$

#### 4.1. SPINE JOINT OPTIMIZATION

We note that  $u_i$ ,  $1 \leq i \leq N - 1$  determines the line segments equations. In other words  $u_i$  are correlated with  $a_i, b_i$ . To obtain this relationship we first write the equation of the  $i^{th}$  polyline as:

$$y_i = f(u_i) + m(x - u_i), \quad m = \frac{f(u_i) - f(u_{i-1})}{u_i - u_{i-1}}. \quad (4.4)$$

where in Eq. 4.4,  $m$  corresponds to the slope of the  $i^{th}$  polyline. From Eq. 4.4,  $a_i, b_i$  can be written as:

$$a_i = m = \frac{f(u_i) - f(u_{i-1})}{u_i - u_{i-1}}, \quad b_i = f(u_i) - \frac{f(u_i) - f(u_{i-1})}{u_i - u_{i-1}}u_i. \quad (4.5)$$

Thus Eq. 4.1 can be rewritten as:

$$F(u_1, \dots, u_{N-1}) = \sum_{j=1}^N \int_{u_{i-1}}^{u_i} (f(x) - (\frac{f(u_i) - f(u_{i-1})}{u_i - u_{i-1}}x + \frac{f(u_i) - f(u_{i-1})}{u_i - u_{i-1}}u_i))^2 dx. \quad (4.6)$$

Obtaining the partial derivative of Eq. 4.6 with respect to  $u_i$ ,  $1 \leq i \leq N - 1$  and set it to zero lead to form the system of equations as follow:

$$\frac{\partial F}{\partial u_i} = 0, \quad 1 \leq i \leq N - 1 \quad (4.7)$$

This system of equation is in general difficult to solve specially when  $N$  increases. We solved the problem by forming the system of equations from Eq. 4.7 for one joint ( $N = 2$ ) however, for larger number of joints it is very difficult to solve. Therefore it seems more rational to try to solve the problem numerically. In the next section, a numerical method would be presented which practically replaces the analytical solution. Although the analytical solution is supposed to be more precise than the numerical approach, the optimization results suggest that the numerical method is an accurate solution.

##### 4.1.2 Spine Optimization - Numerical Approach

As pointed out in the previous section, the objective in the spine optimization problem is to find positions of a fix number of points in a given curve. Let's define a cost function as the sum of areas between the given curve and the approximated polylines (as shown in Fig./reffig:y=Fx) in yellow). The optimal positions of the points can be reached by minimizing this area cost function. Therefore the goal here is to find the positions of such points in the curve that minimize the cost function. This way we know that the polylines are best fitted to the curve.

A heuristic search process would be a good candidate to find the positions of the points and evaluate the area. The minimum area hence corresponds to the optimal solution. One approach is to use the Genetic algorithm. In this sense, the population (chromosomes) is considered as a set of points on the curve and in each

generation the population evolves towards the better solutions. The evolution starts from a randomly generated population and in each iteration of the algorithm, the fitness function for candidate solutions is evaluated. This fitness function is actually the area cost function defined in Eq. 4.3. In fact, this area is the cost of each set of candidates in each generation and we are interested in the minimum cost. Since we know that there exists a minimum area, we know that a satisfactory solution would be reached and the algorithm converges toward the best solution.

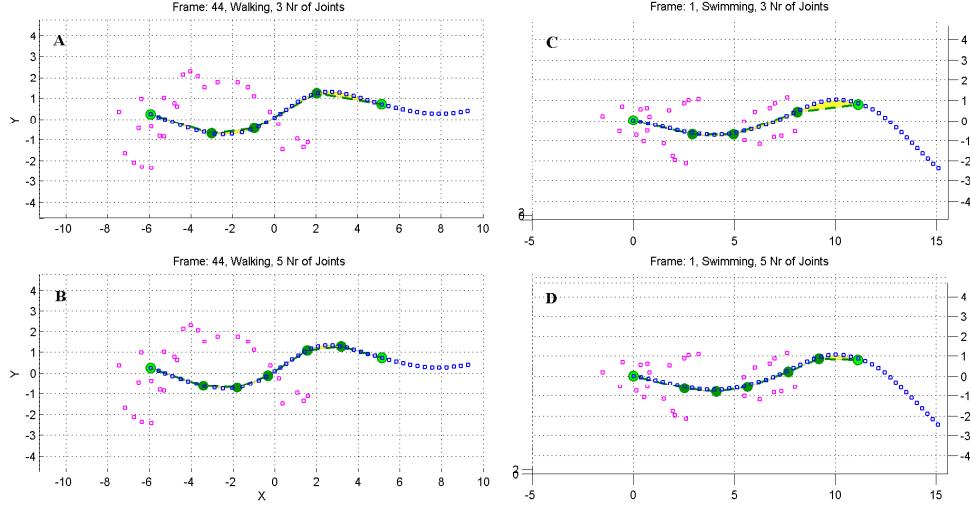
To solve this problem, the MATLAB optimization toolbox, "global optimization" was used. Using the genetic algorithm, the chromosomes were defined as the points in the curve and the cost function defined as the total area between the curve and the polylines. In order to determine the best solution that fits to different gaits of locomotion including swimming and walking, we ran the optimization over all the data in swimming and walking. We chose 6 videos in walking and 6 in swimming. Each video contained one cycle of locomotion, and the optimization were obtained over these 12 videos. We know that during one cycle of locomotion, the spine curvature would be different and the solution should contribute to all curvatures. In each iteration, the algorithm takes into account the solutions that best fit to all curves in all videos. The optimization results intuitively imply that we have reached the optimal positions of the given number of joints that best fits to all possible spine curvatures that the animal employs during walking and swimming. In addition, for the optimization we chose a distinct range on the spine which started from the neck and ended to a point in the tail. The end point was a point in the tail where the tail employed less curvature.

### 4.1.3 Spine Optimization Results

Fig.4.3 indicates the results of the spine optimization for 3 and 5 number of joints on the spine. The results are illustrated both for walking (Fig.4.3 (A), (B)) and also for swimming (Fig.4.3 (C), (D)). In Fig.4.3 spine is illustrated by blue squares, the yellow parts are the area between the curve and the polylines (which are supposed to be minimum), the green lines are the polylines and the dark green circles are the optimal joints on the spine. The figure shows that higher number of points results in better curve fitting.

As pointed out before, the spine optimization problem should address two issues; the number of optimal joints and the positions of them on the spine. So far we have solved the positions problem with the assumption that the number of joints are known in advance. The number of joints should be selected optimally to be compatible with the spine curvatures in different gaits of locomotion, in addition it should address the mechanical constrains. As inferred from the figures 4.3 (A) and (C), 3 joints are not the optimal number to estimate the spine and the number should be increased. However at some level there should be a threshold to pick the optimal number. We performed the optimization from 3 up to 15 number of joints on the spine. Fig. 4.4 shows the error of optimization versus number of joints. The error here is the total area between the spine and the estimated polylines for all

#### 4.1. SPINE JOINT OPTIMIZATION



**Figure 4.3.** Position Optimization results for 3 and 5 number of joints, in the optimization, both walking and swimming data are considered, this figure shows that higher number of points results in better curve fitting. (A) The optimal positions of 3 joints on the spine in one frame of walking, (B) The optimal positions of 5 joints on the spine in one frame of walking, (C) The optimal positions of 3 joints on the spine in one frame of swimming, (D) The optimal positions of 5 joints on the spine in one frame of swimming.

frames of a given video.

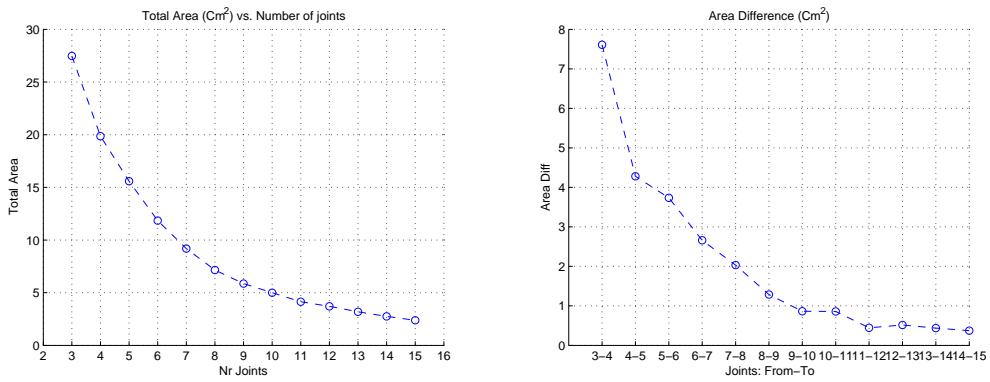
As Fig. 4.4-left implies when the number of joints are increased, the total area will be decreased. Fig. 4.4-right shows the error differences which are obtained from differentiating the left figure. This gives us the criterion to pick the number of joints. According to the figure, we observe that the differences are decreased almost linearly up to joints 9-10. However after this, the the differences will get a constant value. We see that the value of 9-10 compared to the value of 10-11 is almost equal. This suggest that 10 number of joints is a reasonable choice since after 10 joint there is not a remarkable change in the error difference. Consequently, with number 10, we have reached an optimal number of joints in the spine which perfectly represents the spine curvature in different gaits. Recall that only the number of trunk degrees of freedom in the real animal is 48 ( $16 * 3$ ), this number seems a reasonable choice. The spine optimization problem is now solved since both number and the positions of the joints are knowns.

Table 5.3 shows the optimal position of each joint in percentage of the total range of optimization in the spine. Fig.4.5 (A) and (B) shows the final spine joint optimization in walking and swimming respectively. As indicated in the figure, the optimization algorithm put 5 joints in the trunk and 5 in the tail. The results of the optimization implies that the optimized joints are capable of imitating the spine curvature both in swimming and in walking. Note that the range of optimization does not cover the entire tail since the end part of the tail does not employ too much

of curvature. Later in the robot development we will put a compliant material for this part of the tail that is not considered in the optimization.

**Table 4.1.** Optimal position of each joint in percentage of the total range of optimization with the neck as the starting point. As an example, if the neck is the start point, the distance between the neck and joint 1 is 12% of the total range of optimization.

Joint	1	2	3	4	5	6	7	8	9	10
Percentage	12%	23.1%	32.4%	40.8%	49.4%	59.7%	67.9%	74.9%	82.3%	91.3%



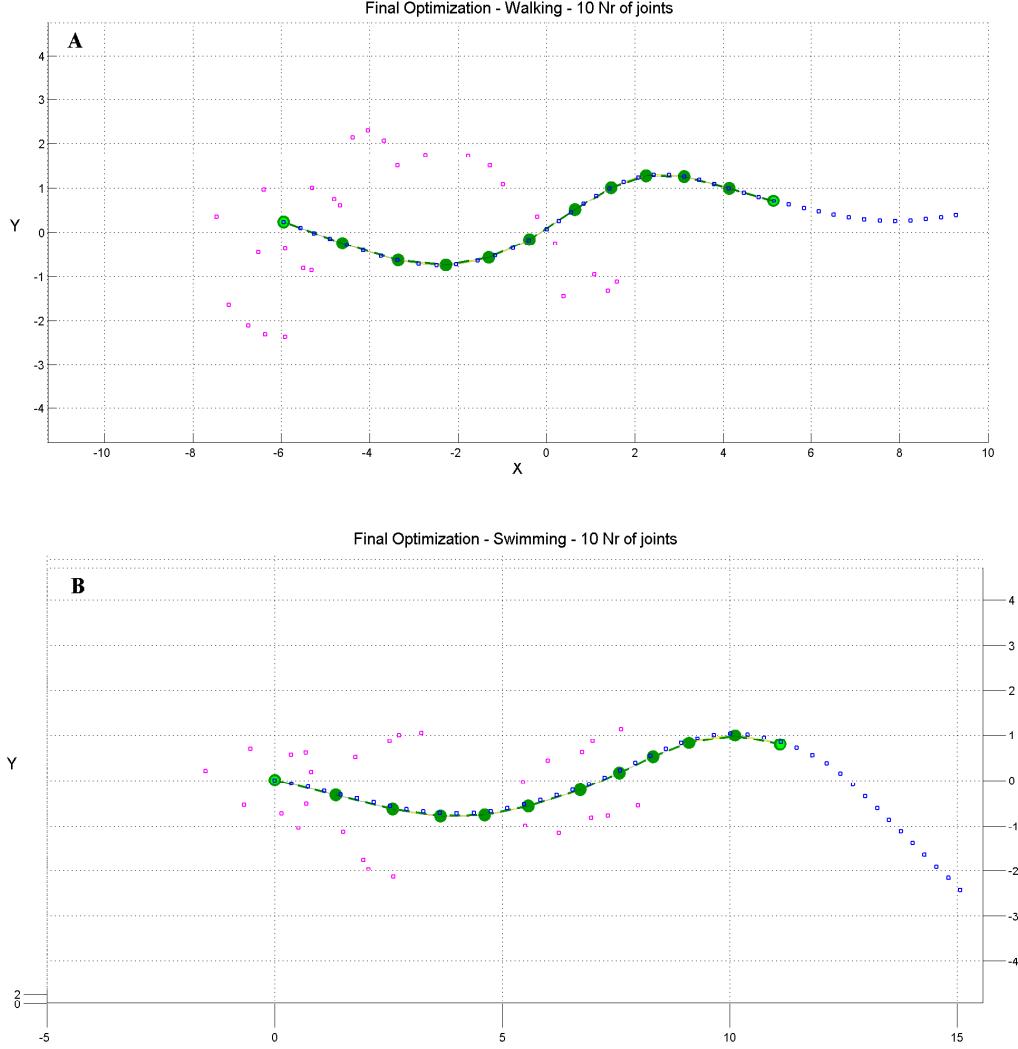
**Figure 4.4.** Optimization for different numbers of joints: Left: Total area of the entire cycle of locomotion, Right: Differences of total areas

## 4.2 Limb Model Simplification

In this section we try to simplify the limbs model in terms of the degrees of freedom. The reasons for minimizing and simplifying the joints are mainly mechanical constraints. Mechanical integration, mass distribution of the robot, and center of mass are some reasons that will be discussed later in chapter 5. According to X-ray videos and also the anatomy, the animal employs 3 degrees of freedom in its shoulder/hip joints during walking gait. In addition 1 degree of freedom is used to the elbow/knee. The palm in the forelimbs and the foot in the hindlimbs consist of an articulated structure which is difficult to describe as distinct degrees of freedom. In chapter 5 we will explain that we use passive joints with compliant materials for the palm/foot.

The focus of the model simplification is therefore on the shoulder/hip joints. In the forelimbs, mechanical integration is not a problem, i.e. we could integrate 3 degrees of freedom in the shoulders of the robot. However this is a big problem in the hindlimbs when it comes to hips. In general it is possible to integrate 3 degrees of freedom in the hips, however the salamander skeleton and anatomy suggest that

## 4.2. LIMB MODEL SIMPLIFICATION



**Figure 4.5.** Final Spine Optimization that fit to swimming and walking, 10 joints are the optimum number, and also their optimal positions are shown. Note that the range of optimization does not cover the entire tail since the end part of the tail does not employ too much of curvature. The figure shows one frame of walking and swimming, as it is clear from the figure, the yellow area is minimized. (A) One frame of walking, (B) One frame of swimming.

the space between the hips are small. This in return means that if we want to integrate a big volume of motors in the hips, the total dimensions of the robot will increase enormously. This is not a problem in the forelimbs since we have enough space between the shoulders according the skeleton model of the animal.

Due to structure of the limb mentioned above, the only degree of freedom that could potentially be simplified is rolling of the hip. Although in the hindlimbs, the hips are exhibiting rolling specially at the beginning of the swing phase, if we

manage to respect the kinematics of the hindlimbs it is possible to neglect this degree of freedom. Considering the hindlimb as a robot arm with the origin placed on the hip, the rolling of the hip could be neglected if the end-effector<sup>2</sup> trajectories could imitate the same trajectories as the animal does. The robot arm model here contains 2 degrees of freedom in the hip<sup>3</sup> and 1 degree of freedom in the knee. Using the forward kinematics of the arm and obtaining the configuration space<sup>4</sup> of the ankle as the end effector, we investigate if the ankle of our arm model could reach the desired trajectories.

### 4.2.1 Hindlimb Forward Kinematics

The goal of this section is to derive the forward kinematics of the hindlimb as a robot arm and try to determine the ankle configuration space as the end-effector. In order to derive the arm forward kinematics we should define appropriate transformation matrices. These matrices gives us the the position of the end-effector as a function of arm angles. Thus the objective here is to find transformation matrices which for a set of arbitrary angles, gives the position of the ankle in the Cartesian space. In general, the transformation matrix in homogenous coordinates is composed of a rotation matrix and a translation matrix:

$$T_{4 \times 4}(\theta) = \begin{pmatrix} R(\theta) & t \\ 0 & 1 \end{pmatrix} \quad (4.8)$$

where in 4.8,  $R_{3 \times 3}$  is the rotation matrix and  $t_{3 \times 1}$  is the translation vector. One thing that makes these transformation matrices so useful the possibility of multiplication of them in a chain to combine several transformations.

Fig. 4.6 shows the structure of our arm model. Note that in the figure, the 3 local frames are separated from each other just to visualize the roll, yaw and pitch rotations in the hip, however these 3 frames should all be placed in the hip joint. In fact, here the translations  $t_0$  and  $t_1$  equal to zero since there is not translation in the hip joint. Note also that the figure only shows one possible order of the degrees of freedom in the hip. The order is roll, yaw and pitch, however other orders could be; (yaw,roll,pitch), (pitch,roll,yaw) and (yaw,pitch,roll). According to the figure, for each joint we define the rotation and translation as follows:

**Hip:** The hip joint consists of three frames as:

Roll: Rotation around local Y-axis - No translation

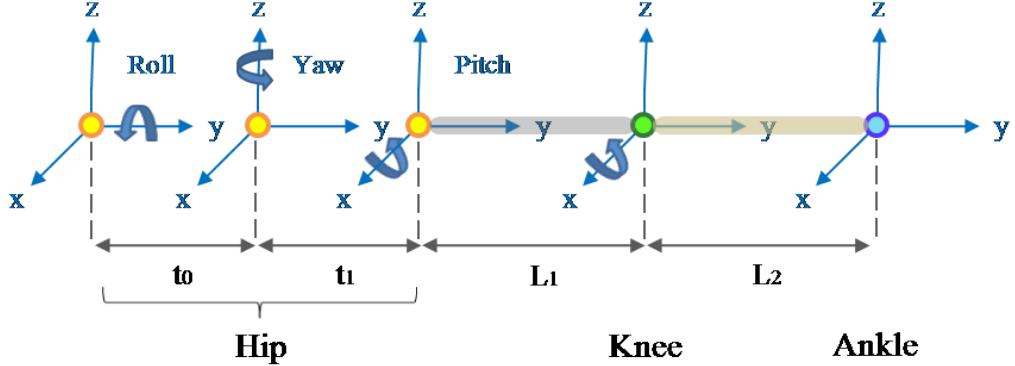
---

<sup>2</sup>ankle is the end -effector

<sup>3</sup>Yaw and pitch: suppose that a right handed frame is placed on the hip, where the limb lies on the Y-axis when all angles are zero, and Z-axis pointing upwards. Yaw is defined as the rotation around Z-axis, and pitch is the rotation around the X-axis.

<sup>4</sup>The reachable space

## 4.2. LIMB MODEL SIMPLIFICATION



**Figure 4.6.** Structure of the arm joints: 3 degrees of freedom in the hip rotating around X-Y-Z axis without any translation among the frames ( $t_0 = t_1 = 0$ ), 1 degree of freedom in the knee rotating around X-axis with  $L_1$  translation, and the ankle as the end-effector with translation  $L_2$  and no rotation

$$T_y(\alpha) = \begin{pmatrix} \cos(\alpha) & 0 & \sin(\alpha) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.9)$$

Yaw: Rotation around local Z-axis - No translation

$$T_z(\beta) = \begin{pmatrix} \cos(\beta) & -\sin(\beta) & 0 & 0 \\ \sin(\beta) & \cos(\beta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.10)$$

Pitch: Rotation around local X-axis - No translation

$$T_x(\gamma) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\gamma) & -\sin(\gamma) & 0 \\ 0 & \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.11)$$

We could determine the transformation matrix of the hip joint by multiplication of 4.9, 4.10 and 4.11. Note that the order of placing roll, pitch and yaw is important

and depending on which order they are placed, we could get different hip transformation matrix. For example one combination could be, roll-yaw-pitch which seems more intuitive compared to other combinations. Note also that in our robot arm model, we considered no rolling in the hip, here we derive the transformation matrix of rolling and later we set this angle to zero (i.e. no rolling):

$$T_{Hip}(\alpha, \beta, \gamma) = T_y(\alpha) \times T_z(\beta) \times T_x(\gamma) \quad (4.12)$$

**Knee:** The knee consists of one frame as:

Pitch: rotation around local X-axis -  $L_1$  translation along Y-axis

$$T_{knee}(\theta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) & L_1 \\ 0 & \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.13)$$

**Ankle:** The ankle is the end-effector that could be defined as:

No rotation -  $L_2$  translation along Y-axis

$$T_{ankle} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & L_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.14)$$

Finally, the transformation matrix from the hip to the ankle is calculated as the multiplication of the chain of the matrices from 4.12, 4.13 and 4.14 as:

$$T = T_{Hip-Ankle} = T_{Hip}(\alpha, \beta, \gamma) \times T_{knee}(\theta) \times T_{ankle} \quad (4.15)$$

Since we assumed that the origin is placed on the hip, we can now calculate the ankle position simply by multiplying the hip coordinate to the final matrix derived in 4.15 as:

$$\bar{X} = \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = T \times \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \quad (4.16)$$

## 4.2. LIMB MODEL SIMPLIFICATION

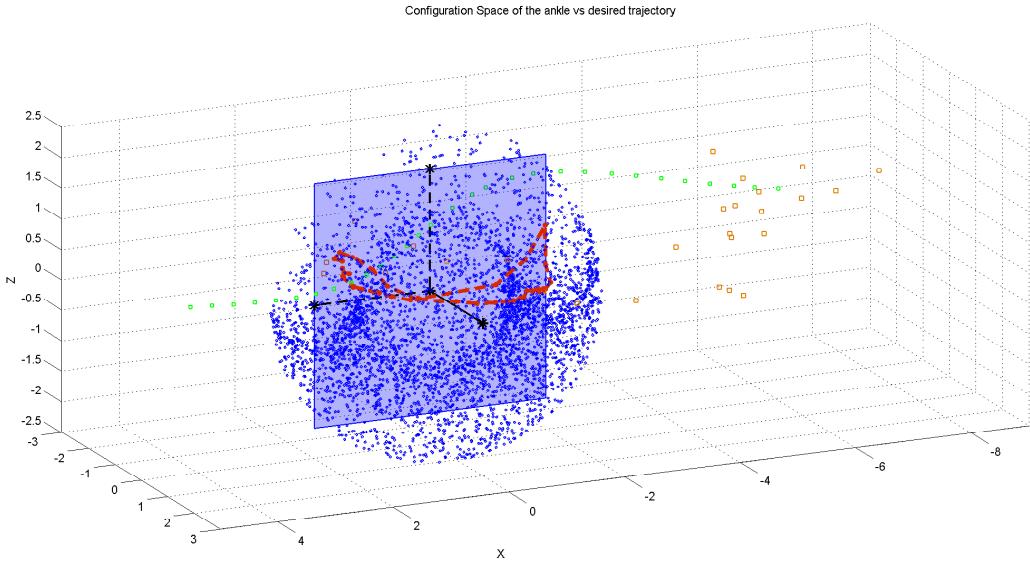
### 4.2.2 Limb Model Simplification Results

Having derived the forward kinematics of the hindlimb, now we could obtain the configuration space of the ankle with different ranges of the joint angles. Table 4.2 shows the ranges of the arm angles.

**Table 4.2.** Angle ranges in the limb used to draw the configuration space of the end-effector, note that there is no rolling.

Angle:	Roll	Pitch	Yaw	Knee
Range (degree):	0:0	-90:90	-90:90	0:180

In order to investigate that the ankle in our robot arm model can reach the ankle trajectories of the animal, we drew the configuration of our model in addition to the trajectories of the animal ankle. A set of uniformly distributed angles (pitch, yaw, knee with zero roll) with the ranges mentioned in Table 4.2 were inserted in our arm model. Fig.4.7 illustrates the configuration space of the end-effector for the right hindlimb. In the figure, the point's cloud of the ankle positions and the desired trajectory are depicted in blue and red respectively. As the figure implies, the cloud which is a spherical volume covers the desired trajectories which means that our arm model without rolling is valid and the hindlimb model can reach the desired kinematics of the animal. This is indeed promising since the hip can be minimized into two degrees of freedom (yaw and pitch), which significantly simplifies the mechanical design and implementation.



**Figure 4.7.** The configuration space of the right hindlimb ankle, The blue point's cloud shows the ankle CS, the red contour is the desired trajectory.



## Chapter 5

# Mechanical Design and Development of the Robot

The mechanical design of the robot plays an important role in the performance of the robot. The mechanical structure should be robust and reliable to avoid inserting uncertainties in the model. In addition the design should be simple to build and assemble. A good design would satisfy the model requirements and provide the necessary angle ranges. In general the design of an articulated body that contains many degrees of freedom is complex. A general overview of the entire design implies that the limb design and the mechanical integration is the tricky part of the design. Also the limbs employ certain angle ranges in terrestrial locomotion while they are kept along the body in swimming. During swimming, the angle ranges in spine are larger than that of ground locomotion. Since the robot is supposed to work both on the ground and in water, the structure should be designed accordingly to achieve the desired angles ranges and be compatible for both swimming and walking. In this chapter we will address the mechanical design of the robot by explaining the important factors involved in the design, we also discuss the design of each part of the robot in detail. The mechanical development of the robot will be mentioned later in this chapter.

### 5.1 Design Tips and Criteria

The robot design was inspired from the skeleton and the anatomy of the real salamander. Fig. 5.1 illustrates the Computed Tomography Scan<sup>1</sup> of the salamander skeleton. Fig. 5.1 (A), (B), and (C) are the dorsal view, ventral, and lateral views respectively. According to the animal skeleton, some key dimensions were extracted as basis for the mechanical design. These data that were obtained from the graphical user interface are shown in Table 5.1. The distance between the shoulders, distance from the neck to the hind girdle (trunk + neck), distance between the

---

<sup>1</sup>CT-scan by Alexander Haas from the university of Hanover and Nadja Schilling.

## CHAPTER 5. MECHANICAL DESIGN AND DEVELOPMENT OF THE ROBOT

hips, and the limb's lengthes are the remarkable dimensions that were considered in the design. In order to replicate the structure of the animal, we tried to respect these dimensions by scaling up the dimensions for the robot design.

**Table 5.1.** Key dimensions extracted from the salamander skeleton.

Part	Length (Cm)
Shoulder's distance	1.76
Forelimb:	
<b>1</b>	1.6
<b>2</b>	0.76
Hips's distance	0.66
Hindlimb:	
<b>1</b>	1.1
<b>2</b>	0.66
Trunk and neck	6.2
Tail	5.6

The scaling factors were determined based on how much we can minimize the size and integrate the mechanics of the joints. The most important factor was the size of the motors. Obviously, the smaller the motor, the smaller the robot size would be since it is easier to integrate the joints. We chose RX-28 Dynamixel servo motor as the main actuators of the robot. The technical specifications of the motor and the motivation of why we selected this kind of motor will be discussed in chapter 7. Fig. 5.2 shows the servo motor and its dimensions in millimeters.

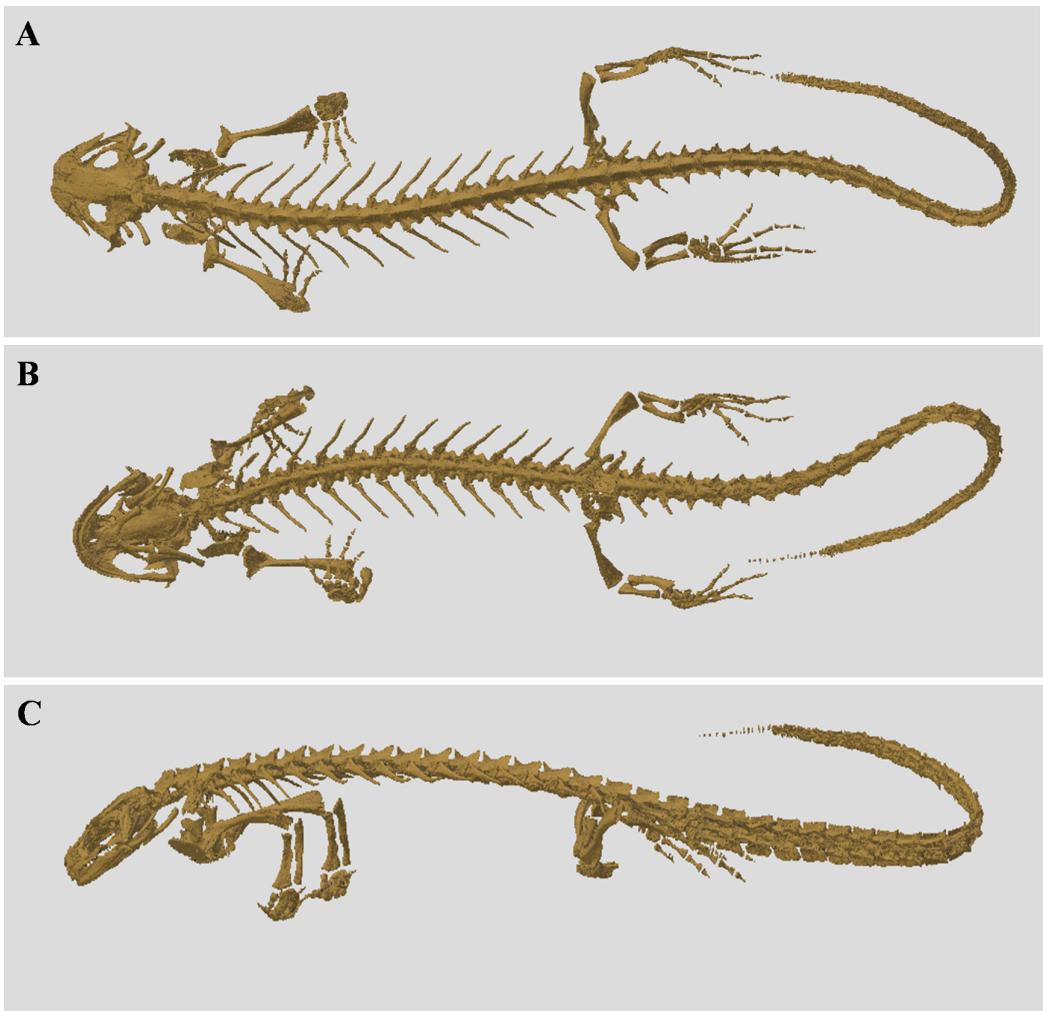
In addition to the motor size, another important factor that affects the robot size is the hindlimbs. As discussed previously, the robot design is based on the dimensions of the animal skeleton. One of them is the distance between the hips. The integration of the motors in the hips of the hindlimbs in order to reach the minimum distance between the hips determines how much we can minimize the robot size. In fact this was the starting point of the design. When the hips were designed, the rest of the body were scaled and designed accordingly. In the forelimbs, the distance between the shoulders are bigger than the distance between the hips. This implies that in forelimb design, we have a larger space to integrate the servo motors.

Apart from the dimensions in the skeleton, the mass distribution and center of mass of the robot should resemble those of the animal so that the robot model would be close to the animal. Fig. 5.3 shows the mass distribution<sup>2</sup> of the animal where the center of mass is shown in black circle. The robot design should satisfy the center of mass and the distribution. This seems to be more important in swimming, thought the walking gait will also be affected by the mass distribution and center of mass. In the following sections, the design of each part of the body including spine,

---

<sup>2</sup>By Konstantinos Karakasiliotis and Jean-Marie Cabelguen

## 5.2. SPINE DESIGN



**Figure 5.1.** The CT Scan of the salamander skeleton, a design inspiration for the robot design; (A) Dorsal view, , (B) Ventral view (C) Lateral view.

hindlimbs and forelimbs will be discussed<sup>3</sup>.

Table 5.2 summarizes the final dimensions of the robot.

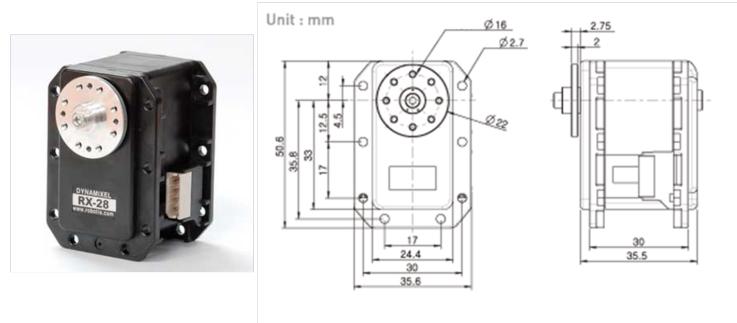
## 5.2 Spine Design

The spine design is the most straight forward design of the robot. Recalling the model optimization in chapter 4, the spine was considered in 2D. In fact the planar spine is a good estimation since the spine angles in the planar X-Y plane are very close to the angles in 3D. Furthermore, in optimization, the spine joints were optimized both for number and the positions of the joints. According to the results of

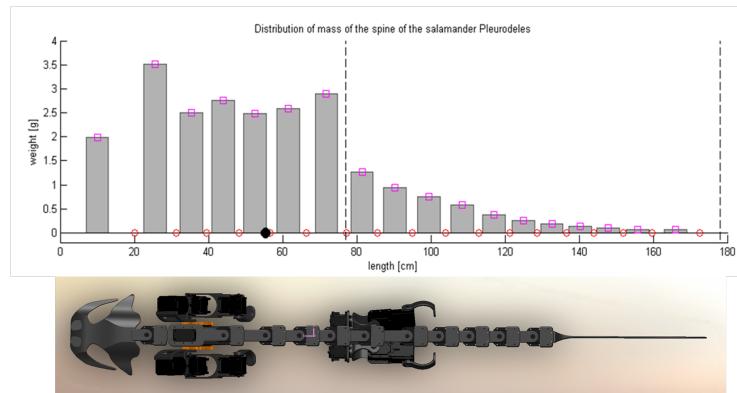
---

<sup>3</sup>The detailed design of the robot was drawn in SOLIDWORKS by Konstantinos Karakasiliotis

## CHAPTER 5. MECHANICAL DESIGN AND DEVELOPMENT OF THE ROBOT



**Figure 5.2.** RX-28 Dynamixel servo motor selected as the main actuator of the robot, Left: RX-28 Dynamixel servo motor, Right: dimensions of the servo motor.



**Figure 5.3.** Top: Mass distribution of the real salamander, Bottom: the robot design in SOLIDWORKS, top view, the design will be discussed later in this chapter.

**Table 5.2.** The final dimensions of the robot.

Part	Length (Cm)
Spine total length	106
Trunk	40
Hip to hip	4.5
Shoulder to shoulder	11.5
Humerus (forelimb)	10
Femur (hindlimb)	7

spine optimization, 10 joints were considered for the spine including trunk and tail. Each joint employs one degree of freedom acting around the Z-axis<sup>4</sup>. In addition one joint is considered for the neck to rotate head.

The results of spine optimization are mentioned in Table 5.3. This table shows

<sup>4</sup>(Yaw, for the definition of yaw refer to Fig. 5.4, Top)

### 5.3. HINDLIMB DESIGN

the optimal position of each joint in percentage of the total range of optimization in the spine. The range of optimization was such that started from the neck and ended in a part in the tail; note that in optimization, the entire spine was not considered. The reason relied on the fact that the end part of the tail did not employ too much of the curvature and therefore the end part of the tail can be replaced by a compliant material. Thus we assumed that the compliant material will be added later to act as the end of the tail and it is not necessary to consider this part in the optimization (Fig. 5.4, Bottom). Having obtained the relative positions of the joints from the optimization, the only objective is to scale them up to get the robot spine. When the size of spine segments were determined, each segment were drawn in SOLIDWORKS.

**Table 5.3.** Percentage of optimal joint's lengths with respect to the total range of optimization with the neck as the starting point. As an example, if the neck is the start point, the distance between the neck and joint 1 is 12% of the total range of optimization.

Joint	1	2	3	4	5	6	7	8	9	10
Percentage	12%	23.1%	32.4%	40.8%	49.4%	59.7%	67.9%	74.9%	82.3%	91.3%

It is worth to mention that the spine model was considered planar with 10 active joints for the spine and 1 for the neck, however in order to lower the center of mass, and also to have a model which is similar to the real animal, 2 passive degrees of freedom<sup>5</sup> were added to the spine so that the spine became 3D. The positions of these 2 passive joints were determined according to the structure of the animal skeleton. Fig. 5.4, Top, schematically illustrates the joints in the spine. In this figure there is also a joint which is so called "scapula" joint. This degree of freedom is considered to rotate the entire fore-limbs. The passive joints will later be equipped with appropriate springs to hold the tail in the desired posture which is close the to animal anatomy. Fig. 5.4, Bottom (A-C) indicates the assembled spine of the robot drawn in SOLIDWORKS. The passive joints of the spine is well shown in the lateral view. Also the end part of the tail is consist of a compliant material.

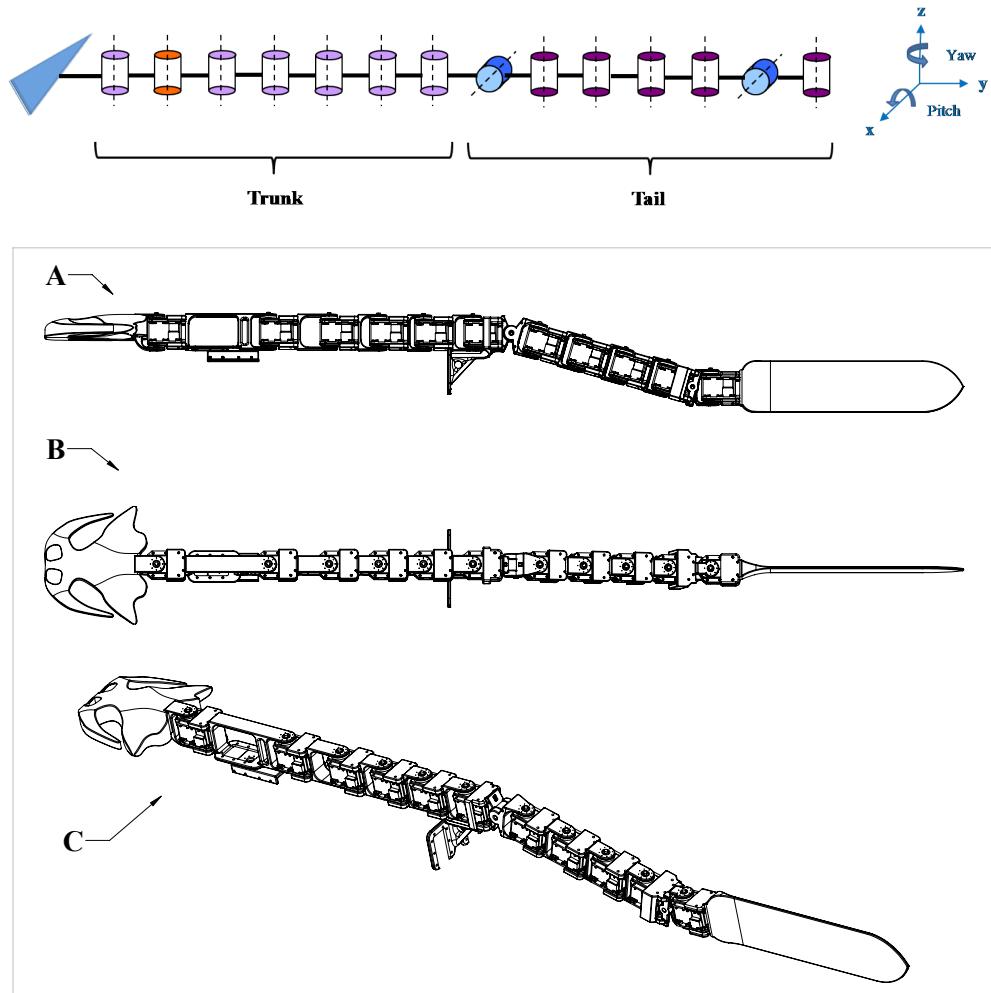
## 5.3 Hindlimb Design

According to the results of the hindlimb model simplification discussed in chapter 4, the hindlimb model was minimized to 3 degrees of freedom; 2 in the hip 1 as the knee. In fact in the simplification, the rolling (rotation around Y-axis) was eliminated since we can reach the desired trajectory of the ankle by pitching (rotation around X-axis) and yawing (rotation around Z-axis). Fig. 5.5, Top indicates the schematic degrees of freedom of the hindlimb. As shown in the figure, the hip employs pitch and yaw rotating around the local X and Y axis respectively. Also the knee has

---

<sup>5</sup>(Pitching: rotation around X-axis, for the definition of pitch refer to Fig. 5.4, Top)

CHAPTER 5. MECHANICAL DESIGN AND DEVELOPMENT OF THE ROBOT

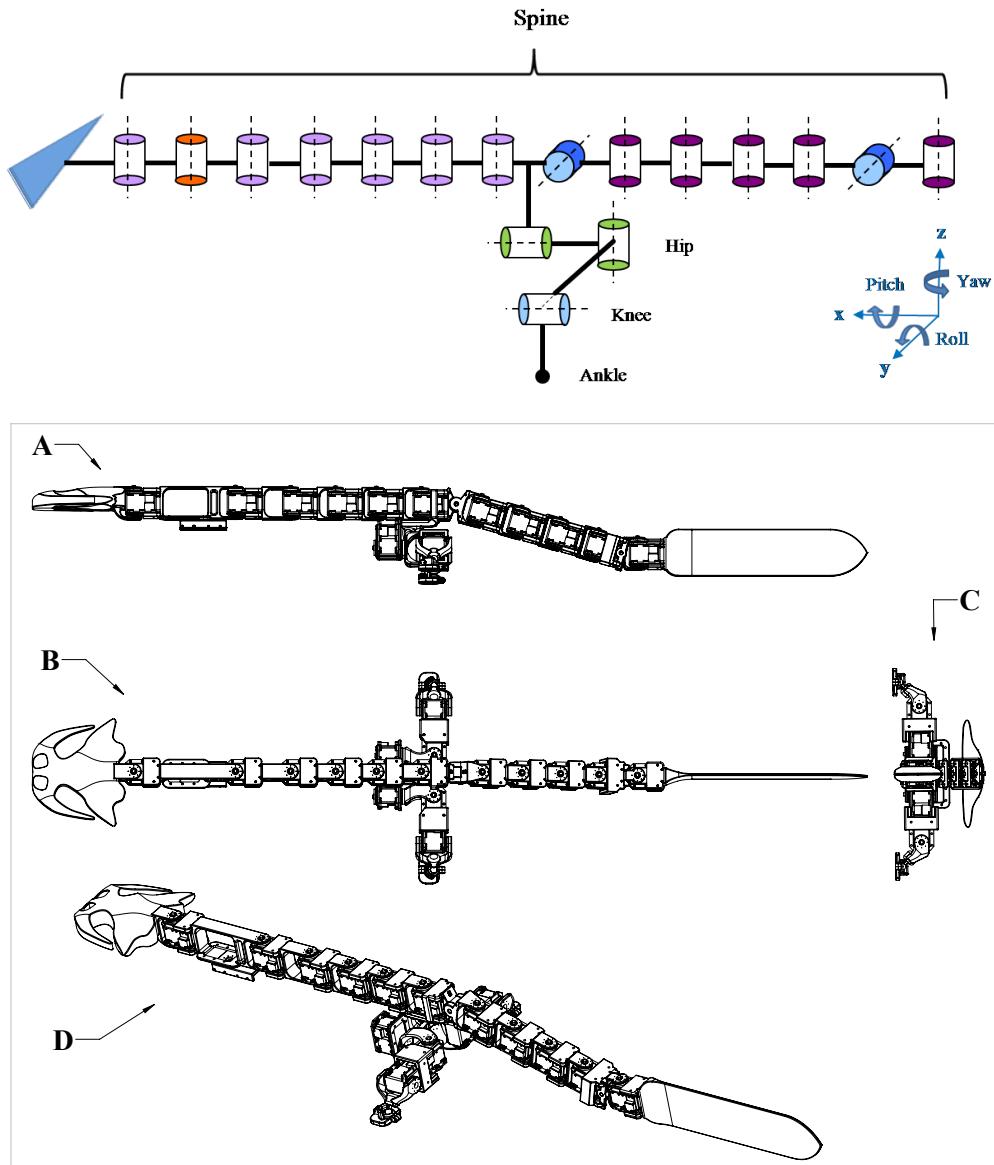


**Figure 5.4.** Top: Schematic degrees of freedom in the spine; 6 active joints in trunk shown in violet, 1 joint for the scapula shown in red, 5 active joints in tail shown in dark violet, and 2 passive joints in the tail shown in blue. Bottom: Drawing of the robot spine in SOLIDWORKS, note that the end part of the tail is consist of a compliant material; (A) Lateral view, (B) Dorsal view, and (C) 3D view.

one pitching degree of freedom rotating around the local X-axis. The ankle joint is considered as a passive joint without any motor. The idea is to use a ball joint so that the ankle can rotate omni-directionally. The joint will later be equipped with some springs to orient the foot appropriately. Fig. 5.5, Bottom (A-D) shows the

### 5.3. HINDLIMB DESIGN

hindlimb drawing in SOLIDWORKS with different views.



**Figure 5.5.** Top: Schematic degrees of freedom of the left hindlimb; 2 active joints in the hip shown in green, and 1 active joint in the knee shown in blue. The ankle uses a passive joint. Bottom: Robot drawing in SOLIDWORKS including spine and hindlimbs; (A) Lateral view, (B) Dorsal view, (C) Backward view, and (D) 3D view.

The foot design is one of the most complex parts of the robot design. In swimming the salamander keep its limbs along the body and thus they are not used.

However in terrestrial locomotion the limbs specially the hindlimbs functionalities are remarkable. In the stance phase the foot is placed on the ground and the whole limb rotates on the foot. The range of rotation is approximately  $100^\circ$ . During the rotation of the limb in stance phase, the foot and the fingers are fixed to the ground. This implies that the foot and fingers should have enough friction with the ground to avoid slipping. When the rotation goes towards the end, the limb steps on the fingers. The fingers gradually lose contact with the ground and the swing phase will be started. In this stage the limb is lifted and the foot is oriented to its initial posture so that the foot is prepared for the next stance phase.

In general it is really complicated to replicate what the salamander does in its feet. We designed the spherical ball joints as the ankle of the robot and used some springs in the foot to reorient the foot in the swing phase. Also the foot was equipped by compliant fingers to have a better contraction with the ground.

## 5.4 Forelimb Design

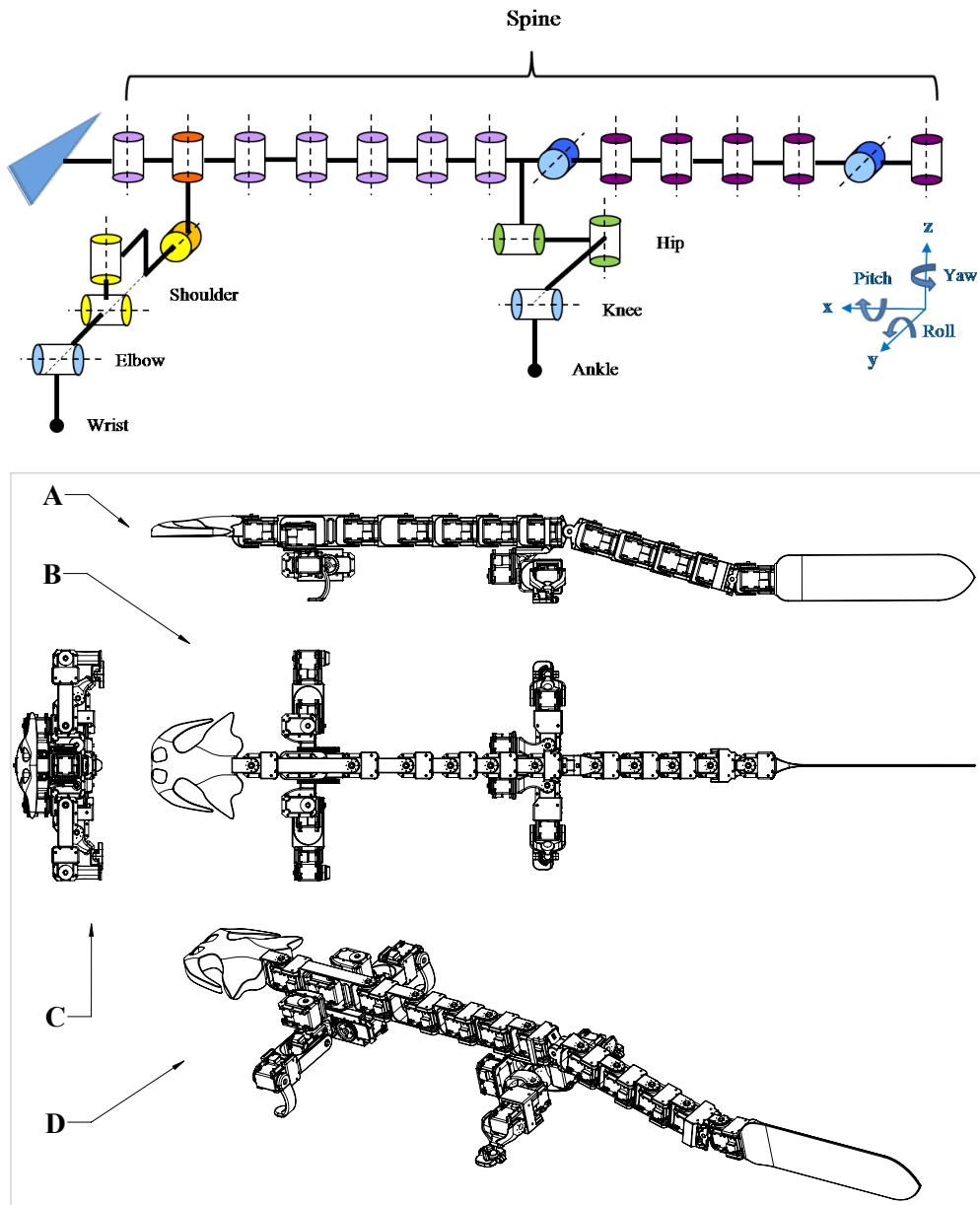
In the design of forelimbs unlike the hindlimbs no joint simplification was used. The reason that one degree of freedom was eliminated from the hip, as explained, was the mechanical problems such as integration, mass distribution and the position of the center of mass. However, as the salamander skeleton implies, we have a larger volume between the shoulders and this is enough to integrate 3 degrees of freedom in the shoulders. In addition, we considered 1 degree of freedom for the elbow and one passive joint in the wrist (similar to the ankle). In total 4 degrees of freedom were considered for each forelimb.

As mentioned in the "Spine Design" section, 1 joint was considered as the scapula to rotate the entire forelimbs. This degree of freedom is acting as yaw and rotating around the Z-axis. The summary of the joints and degrees of freedom is illustrated schematically in Fig. 5.6, Top. The scapula which is indicated in red is placed in the top of the limbs. The shoulder shown in yellow employs 3 degrees of freedom as roll, yaw and pitch, and the elbow that is shown in blue has pitching. Fig. 5.6, Bottom (A-D) shows the entire robot design in SOLIDWORKS including the forelimb design with different views.

Note that we can have different sequences of roll, yaw and pitch. The order of this sequence affects the forward kinematics of the limb. In terms of configuration space (the reachable space) of the wrist as an end-effector, the best order is roll, yaw and pitch which was chosen for the shoulder. This order is also easier to design and implement and more intuitive in terms of placement.

The palm of the forelimbs are quite similar to the foot in the hindlimbs. According to the analysis, it seems as if the palms are more responsible to keep the balance during terrestrial locomotion. Similar to the foot design, the palm design should contain some springs to orient the hands as expected.

## 5.5. ROBOT DEVELOPMENT



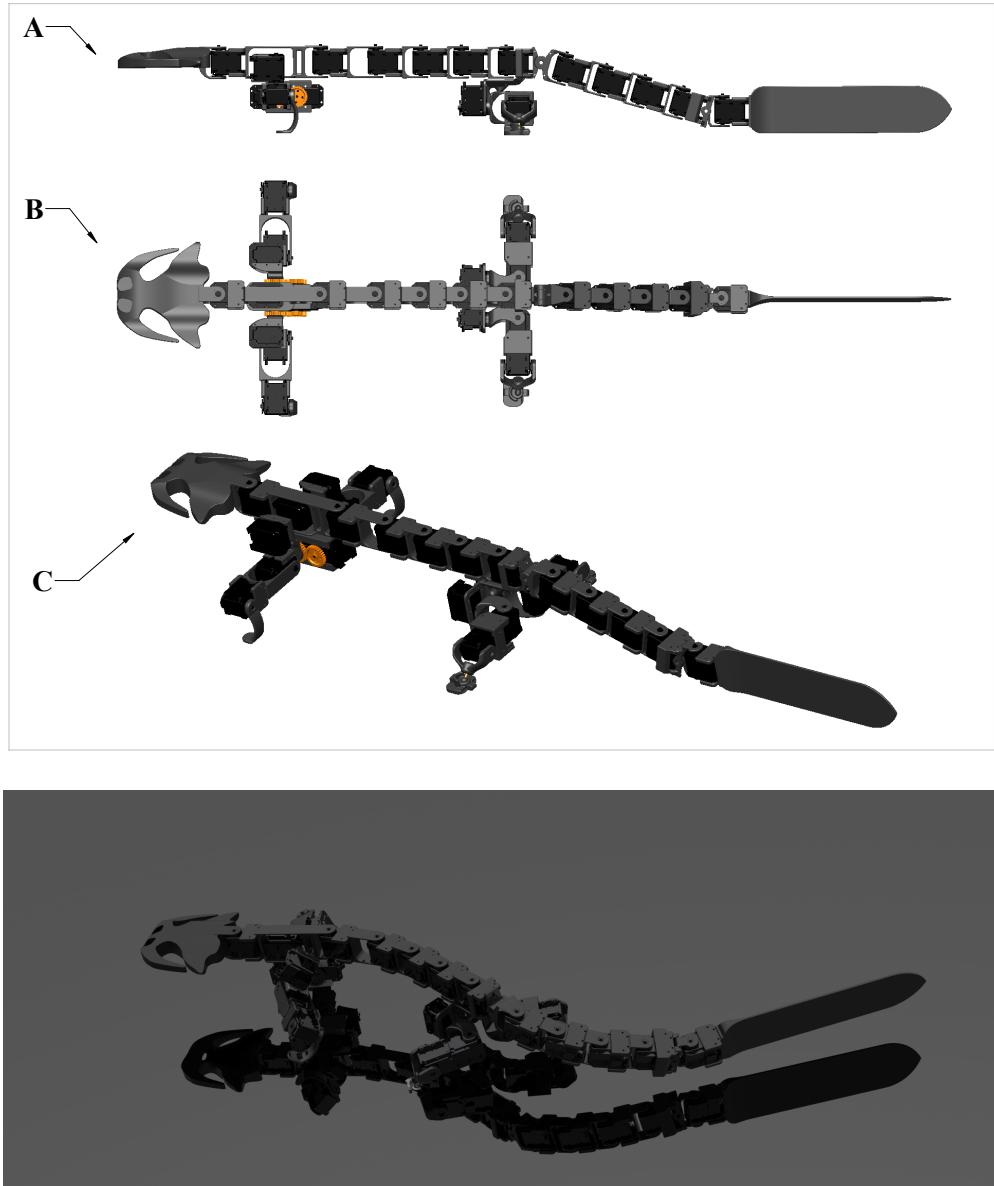
**Figure 5.6.** Top: Schematic degrees of freedom of the forelimb: 3 active joints in the shoulder shown in yellow, and 1 active joint in the elbow shown in blue. The wrist uses a passive joint. Bottom: Entire robot drawing including the forelimbs in SOLIDWORKS; (A) Lateral view, (B) Dorsal view, (C) Front view, and (D) 3D view.

## 5.5 Robot Development

Having designed all parts of the robot, the robot was assembled in SOLIDWORKS. In the design, the ranges of the angles were also considered so that the design would

## CHAPTER 5. MECHANICAL DESIGN AND DEVELOPMENT OF THE ROBOT

address the appropriate ranges for all of the degrees of freedom in different gaits (terrestrial and aquatic locomotion). Fig. 5.7 shows the final design of the robot in SOLIDWORKS.

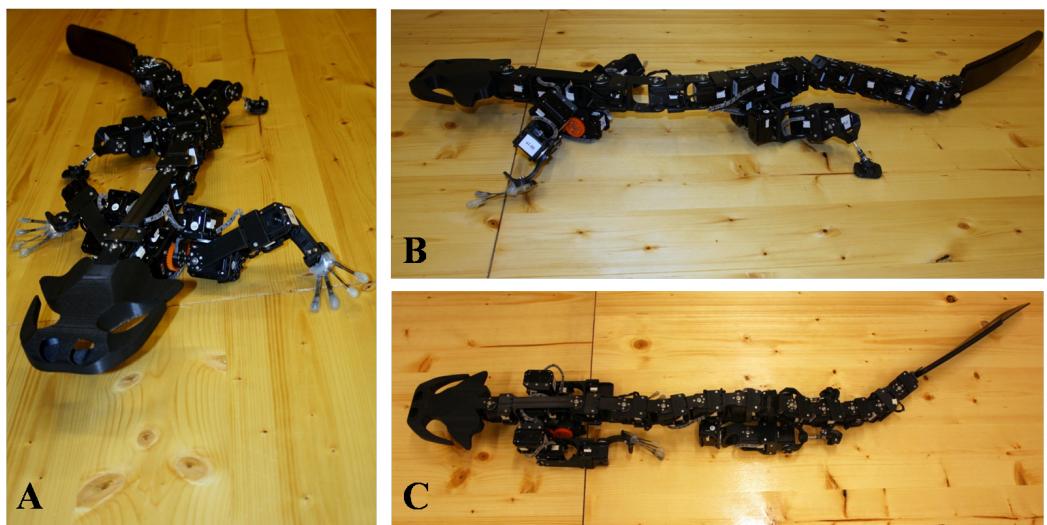


**Figure 5.7.** Robot assembly in SOLIDWORKS, Top: different views of the robot, (A) Lateral view, (B) Dorsal view, and (C) 3D view. Bottom: The walking posture of the robot in SOLIDWORKS.

The detailed design of the robot was a difficult, however in terms of mechanics,

## 5.5. ROBOT DEVELOPMENT

the robot development was straightforward. When all parts of the robot were designed, each part was sent to a 3D printing machine. The machine is a 3D plastic printer which is capable of printing any 3D part drawn as a CAD file. When the parts were ready, we assembled the robot in a short period of time. Fig. 5.8 (A) and (B) shows the final robot assembly in the walking posture and Fig. 5.8 (C) the swimming posture.



**Figure 5.8.** The final robot assembly, (A) and (B) walking posture, (C) Swimming posture.



## Chapter 6

# Physical Simulation

Physical simulation of the model is an essential step that steers the project towards the final development of the robot. The simulation can support the robot model as a true and valid model or simply falsify the modeling. Furthermore, we can investigate the kinematics obtained from the salamander in simulation. The simulation can provide a simple test bed and platform for testing different gaits of locomotion before implementing on the robot. Although there are always some uncertainties and unmodeled factors that reduce the accuracy of the model, the simulated model could be good representative of the real model. Moreover there are remarkable and useful data which can be deduced from the simulation; forces on the structure and joints, required torques for each joint, accelerations, and even model drawbacks and physical analysis for instance. However these issues will not be discussed in this chapter.

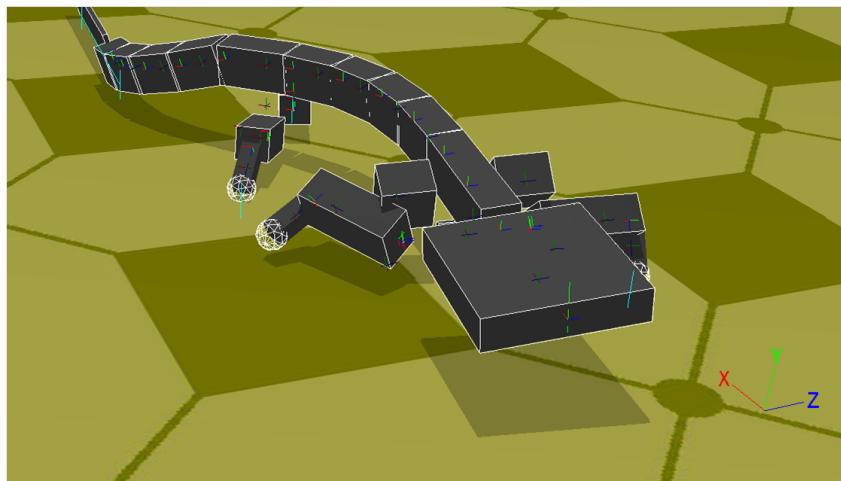
As mentioned previously, salamanders are both capable of aquatic and terrestrial locomotion. On ground they use their limbs and spine to walk while in swimming they only use their spine and the limbs are kept along the body. However, here we only intend to address the ground locomotion of the model for the walking gait, though it is also possible to further extend the model to be used for other gaits such as swimming by changing the environment and other parameters. The simulated model aims to test and verify the kinematics obtained from the salamander skeleton. Before developing the mechanical structure of the robot, it is useful to employ the kinematics to the simulator and test the gait.

In this chapter we will address the simulation of the robot to test the terrestrial locomotion; walking gait, and demonstrate that our robot model is a good model for locomotion. At the end of the chapter, the simulation results will be provided which support our model validity.

## 6.1 Model Simulation in Webots

To investigate the locomotion of the salamander, a robot model was simulated<sup>1</sup> in Webots. The model composed of 26 joints, 11 joints on the spine (trunk and tail) and 14 joints in the limbs. Similar to the mechanical design, there are 11 active joints on the spine and 2 passive joints to lower the tail. Also there are 3 degrees of freedom in each shoulder and 1 in each elbow. In addition, similar to the design, 2 degrees of freedom is considered for each hip and 1 for each knee. However, we did not consider the passive ankle/wrist joints in the simulation, i.e. we assumed that we have rigid spheres with appropriate friction instead of passive foot/palm and fingers. In chapter 5, we designed the mechanics of the robot. In simulation we use the exact segment lengths as we developed in the mechanical design, e.g. the optimized spine lengths, the dimensions of the limbs and so on. Fig.6.1 shows the simulated model in Webots.

Having simulated the model, we controlled the model using C++ language. The developed graphical user interface enables the user to select any X-ray video and create the appropriate kinematics to be used in the simulator. The user can select the intended video in MATLAB and save the kinematics automatically to be applied for the model. Then the only thing to do it to run the simulation in Webots. This made the gait testing significantly easier. We also smoothed kinematics in Matlab to have a smooth behavior in the simulator. We used some conventions to address the angle definition, cycle repetition, etc, which will be discussed in chapter 7.



**Figure 6.1.** The simulated robot model in Webots, 11 joints on the spine, 3 joints in each shoulder, 1 joint in each elbow, 2 joint in each hip, 1 joint for each knee, however the foot/palm and fingers are not simulated, instead rigid spheres are used.

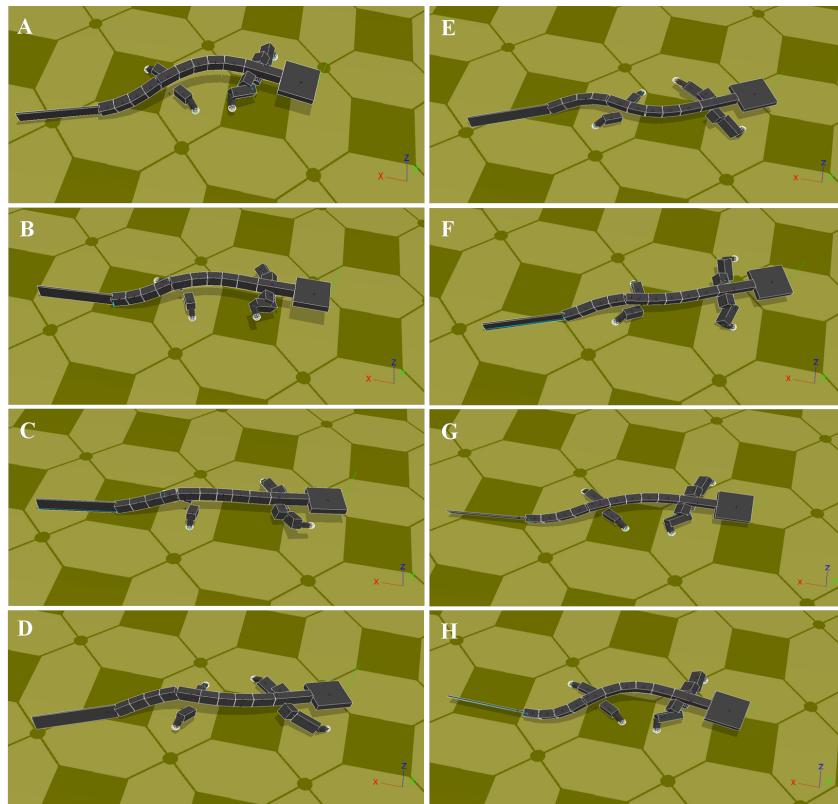
---

<sup>1</sup>By Konstantinos Karakasiliotis

## 6.1. MODEL SIMULATION IN WEBOTS

### 6.1.1 Results

The simulation results of the walking gait indicated that our model is a valid model of the real salamander which can represent the kinematics of the animal. The model could easily walk in the simulator with the kinematics of the real salamander. This verifies the correctness of the kinematics obtained from the animal and also gives support for the model as a valid one. We also tested the backward locomotion by reversing the cycle of locomotion and also the turning by offsetting the spine angles. In both cases the model can perform the tasks also these gaits were not obtained from the kinematics of the salamander. Fig.6.2 (A)-(H) illustrates the walking snapshots of the model in the simulator. The walking frequency is **0.6 Hz**.



**Figure 6.2.** Snapshots of walking gait, (A)-(H) represent different snapshots of one cycle of locomotion.



## Chapter 7

# Robot's Hardware and Control

This chapter describes the hardware and control method that we used for the robot. The type of the servo motors used in the robot in addition to their technical specifications would also be addressed. Furthermore how to control servo motors of the robot, the control board and also how to replicate the salamander kinematics will be discussed. In the mechanical structure of the robot, 26 servo motors were used. We had to used an appropriate hardware to control these motors simultaneously to replicate the desired trajectories drawn from the kinematics. In addition, it is more likely that the motors have a low power consumption with high torque so that later the robot could be equipped with some batteries to become autonomous. Since 25 motors are involved in locomotion (1 motor for the head), they should have enough torque and speed to replicate different gaits (e.g. swimming and walking). In the following sections the hardware specification and also control strategy will be discussed in detail.

### 7.1 Robot's Actuators

As mentioned in chapter 5 (Mechanical Design and Development of the Robot) we used "RX-28 Dynamixel" servo motor as the actuators of the joints. The servo module consists of a DC motor with a reduction gearbox, driver, controller and a network. The servo motors share a common communication and power BUS in the network and it is easy to control several motors in the common BUS when they are connected in a sequence chain. Also the wiring of several servos are convenient in that they are connected serially as "daisy chain"<sup>1</sup>. We can assign a unique ID (0 ~ 253) for each motor and send the communication packet on the common BUS to control several motors. Fig.7.1 schematically shows several motors connected serially in the common BUS.

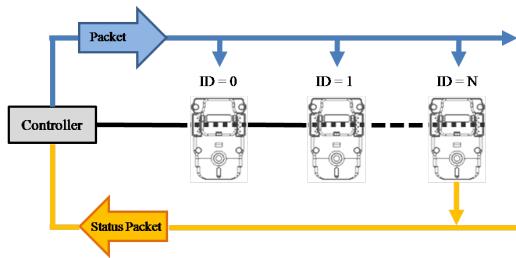
The RX-28 servo motor employs several feedbacks such as position, load, speed, voltage and temperature in the network which reduces the resources. There is a return packet from the motor called status packet which contains the feedback data

---

<sup>1</sup>A wiring scheme in which multiple devices are wired together in a sequence.

and the status of the motor. Endless Turn is another property of the motor which makes it possible to used the motor as a wheel. This property is not applicable for our robot, however it might be useful for some applications.

It is possible to control the speed and the position of the motor with the resolution of (1/1024). In addition, manipulating the output torque pattern of the motor provides the interesting property of elasticity that might be useful in some applications where shock absorption, smooth motion and compliancy is needed. Table 7.1 indicates the specifications of the servo motor.



**Figure 7.1.** The Communication Network of the RX-28 Dynamixel servo motor; Each motor has a unique ID, to control several motors, one packet is sent from the controller to the common BUS while all motors all listening. The motor can feed the status packet back to the controller to analyze the feedback information.

The RX-28 communication protocol is an asynchronous serial communication with 8 bit, 1 Stop bit, and None Parity. We have different protocols when controlling a single motor or several motors. As an example, Fig.7.2 briefly mentions the communication protocol when controlling one motor. The first two hexadecimal numbers indicated the starting of the packet, "ID" is the ID number of the motor, "LENGTH" is the packet length, "INSTRUCTION" is the command that can be read/write, "PARAMETERS" are the values of the data, and finally "CHECK SUM" is used to verify that the packet is not damaged.



**Figure 7.2.** The Communication Protocol; Start of packet with two hex number , ID of motor, Length of packet, Instruction W/R, Data Parameters and Check SUM to check the correctness of the packet.

## 7.2 Robot's Control Hardware

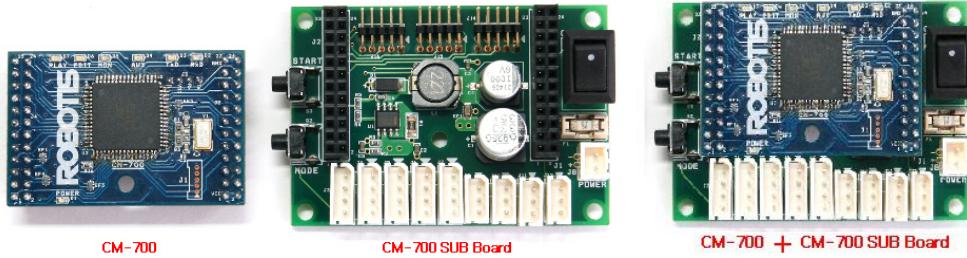
A suitable hardware should be capable of controlling 26 motors at the same time. We used a hardware module as the hardware that consists of two boards. The main board called "CM-700" is a controller with a CPU (ATMega2561 micro-controller), TTL RS485 communication circuit and ZIG-110 connectors. The other board called

## 7.2. ROBOT'S CONTROL HARDWARE

**Table 7.1.** Specifications of RX-28 Dynamixel servo motor.

Weight (g)	72	
Dimensions (mm)	$35.6 \times 50.6 \times 35.5$	
Gearbox Ratio	1/193	
DC Motor	Maxon RE-MAX	
Max Current (mA)	1200	
Standby Current (mA)	50	
Applied Voltage (V)	12	16
Stopping Torque (kgf.cm)	28.3	37.7
Speed (Sec/60 degrees)	0.167	0.126

”CM-700 SUB” composed of a power department, the connector department, a switch, and an additional circuit for 5 pin peripheral devices. The CM-700 controller board is mounted on the CM-700 SUB board and the entire module can easily control several motors. Fig.7.3 illustrates the boards separately and the whole hardware module together. CM-700 controller board connects serially to the PC and the ATMega2561 micro-controller is programmed using C language. The small dimensions of the hardware module make it possible to put it inside the head of the robot.



**Figure 7.3.** The Hardware Module; from the left: CM-700 controller, CM-700 Sub board, and the two boards together as the entire hardware module.

Since the servo motors are sharing a common BUS for communication and power, to control several motors that are connected as a daisy chain, it is only necessary to connect one motor from the chain to the controller and send the packet from the controller to the BUS. Table 7.2 shows the specifications of the hardware. According to the Table 7.2 and Table 7.1, the common voltage range of the entire hardware including the boards and the motors is  $12V \sim 16V$  and to energize the entire robot it is only necessary to connect a power cable to the board. Alternatively, since the servo motors are sharing a common power BUS, we can connect the power cable to any of the motors in the chain. The most convenient was to connect the power to the last motor of the tail.

**Table 7.2.** hardware specifications.

Weight (g)	37.3
<b>Dimensions (mm)</b>	55 × 40 × 15
Micro-Controller	ATMega2561
Applied Voltage (V)	7 ~ 35
Current (A)	0.04 ~ 0.9
I/O Device	4-Pin Connector for Dynamixel 3-Pin Connector for Dynamixel 4-Pin Serial to PC communication 5-Pin I/O for Peripheral-Devices

### 7.3 Robot Control

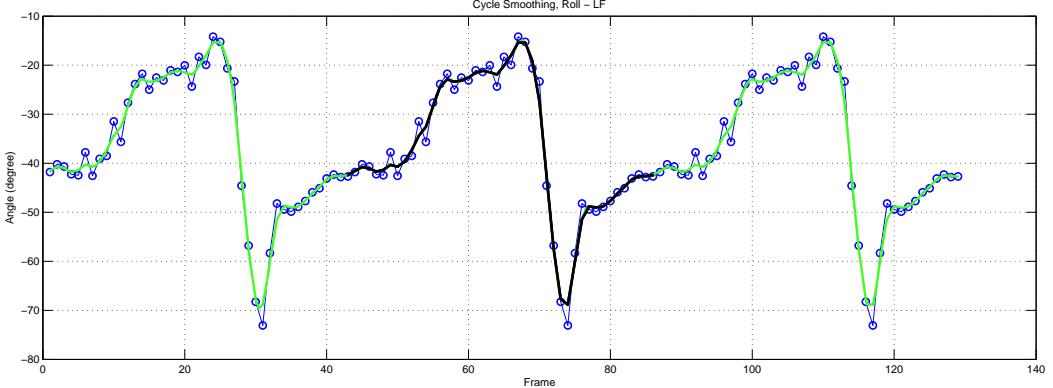
To control 26 servo motors used in the structure of the robot, a unique ID was assigned to each motor. The IDs can have the range from 0 to 255. We started from the neck servo with ID equals to zero up to right hindlimb knee servo with ID equal to 25. Table 7.3 shows the number of IDs assigned to the servo motors.

**Table 7.3.** Assigned IDs to each RX-28 Dynamixel servo motor.

ID Number	0	1 - 10	11	12 - 15	16 - 19	20 - 22	23 - 25
Motors	Neck	Spine	Scapula	Left ForeLimb	Right ForeLimb	Left HindLimb	Right HindLimb

The hardware module is connected serially to the PC and the controller is programmed using C language. We chose one walking X-ray Video which was a good representative of the animal locomotion and at the same time contained a full repeatable cycle of locomotion. A cycle of locomotion is defined as the time of one stride to the next following stride. Stride is the time from one hindlimb (left/right) contact with the ground to the subsequent contact. To replicate the kinematics data obtained from the video, we prepared some lookup tables containing the angles of each joint in one cycle of locomotion. The lookup tables are some matrices that have the angles of all of the joints in one cycle. In order to have a continuous locomotion, the cycle should be repeated in a loop and the cycle change should be smooth. This means that the angles of each particular joint should have similar values at the beginning and at the end of the cycle. To reach this goal, each joint angle was plotted in three cycles and a spline was fitted to this plot. Then the middle cycle was selected and the angle's values were interpolated as the final values for the lookup table. Selecting the middle cycle ensures that the beginning and the end values are smoothed and have similar levels. While fitting the spline to the curves, we also smoothed the values to reduce the noise and have a smooth behavior in the motors. Fig. 7.4 indicates an example of the cycle smoothing in Left Forelimb-rolling angle. Note that the X axis represents the frames of the cycle (order of time) and the Y axis the values of the angles.

### 7.3. ROBOT CONTROL



**Figure 7.4.** Cycle Smoothing, this is to ensure that the changes between the cycles are not abrupt, instead the cycle change is smoothed. Blue: Left forelimb rolling actual values of the angle in 3 cycles, Green: the fitted spline to the curve with a certain smoothing factor, Black: the interpolated angle selected for the lookup table.

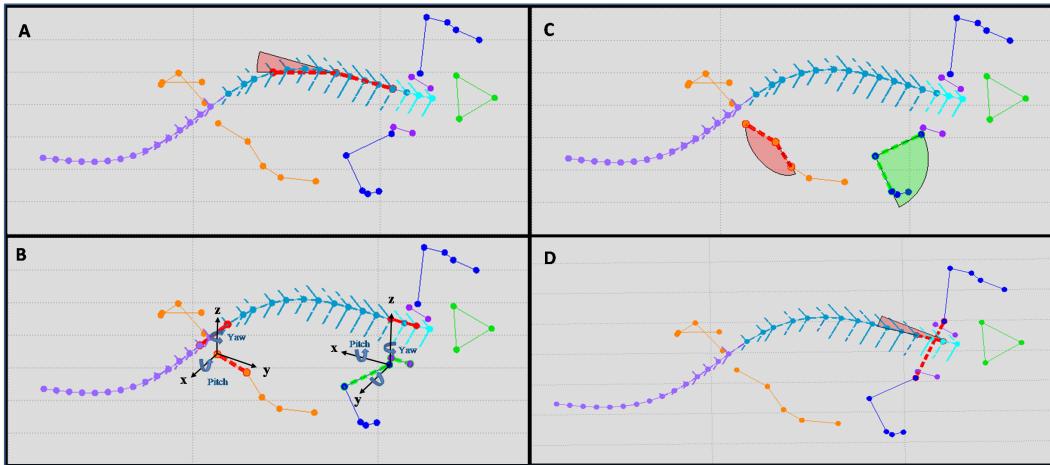
#### 7.3.1 Angles and Definitions

To have convenient rules for robot control, we used some conventions for the definition of the angles similar to chapter 3. The following definitions were used in the control part of the robot:

- **Spine angle:** is defined as the angle between the lines connecting the adjacent pairs of spine joints (as shown in Fig. 7.5 (A), red angle). Zero angle happens when the two lines are along each other, the figure shows a negative angle. Since in spine optimization the spine model was considered in 2D, here the spine angles were calculated also in 2D.
- **Forelimb angles:**
  - The shoulder joint has 3 degrees of freedom; "roll", "yaw" and "pitch". Putting a local right-handed frame on the shoulder which the arm lies on the Y-axis, the roll is defined as the rotation around the Y-axis. The frame is put such that the X-Y plane is parallel to the ground and X-axis is parallel to a reference in the spine (Fig. 7.5 (B)). The reference is defined by the second and the fourth joint of the spine as shown in red line. Using the right-hand rule, if the thumb pointing away from the origin along the Y-axis, the curvature of the fingers indicates the positive rotation of the angle. With similar definitions, yaw and pitch are defined as rotations around Z and X axis respectively (Fig. 7.5 (B)).
  - **Elbow angle:** is defined as the angle between the line connecting the shoulder to elbow and the line of elbow to wrist. The line of shoulder-elbow is considered to be the reference and the zero angle happens when elbow-wrist line lies on this reference (Fig. 7.5 (C), green angle).

- **Hindlimb angles:**

- As we modeled the robot, the hip joint only contains two degrees of freedom, yaw and pitch. Putting a local frame on the hip which the arm lies on the Y-axis, the yaw and pitch are defined as the rotation around Z and X-axis (Fig. 7.5 (B)). The frame is located such that the X-Y plane is always parallel to the ground and X-axis is parallel to a reference in the spine. The reference is defined by 2 joints on the spine; 1 before the hind girdle and 1 after. The right hand rule, determines the sign of the angles.
- **Knee angle:** is defined as the angle between the lines of hip-knee and knee-ankle. All other definitions match that of the "elbow angle" (Fig. 7.5 (C), red angle).
- **Scapula angle:** is defined as the angle between the midline formed by a reference on the spine, with the line perpendicular to the shoulders midline as shown in Fig. 7.5 (D) in red. The reference here is exactly the same reference used in the shoulder angles.



**Figure 7.5.** Different angle definitions. (A) The angle on the spine, choosing 3 adjacent points, the angle is defined as the angle between the lines formed by the pairs of points in the spine. (B) Angles of the shoulder, roll, pitch and yaw, and also the angle of the hip; yaw and pitch. Roll, yaw and pitch are defined as the rotation around Y, Z and X axes respectively. (C) Elbow and Knee angles shown in green and red respectively, the angles are zero when the lines forming the angle are superimpose and  $180^\circ$  when they are in a straight line. (D) Scapula angle, this angle is defined as the angle between the reference (red line) in the spine and the line perpendicular to the midline of shoulders.

### 7.3. ROBOT CONTROL

#### 7.3.2 Kinematic Calculations for Robot Control

After the definitions of the angles, we calculated the angles required for the robot control. The angles of the spine, forelimbs, hindlimbs and the angle of the scapular are the angles we calculated and saved as lookup tables in the robot hardware. All the calculations of these angles were performed in the developed graphical user interface. The user can select any video and save the angles for the robot control. The angles of the spine and the scapula is straightforward to calculate, however we have to use the inverse kinematic problem to calculate the shoulder/hip angles that is discussed in this section.

The angles of forelimb were calculated using the inverse kinematics problem. Given the positions of the elbow and the wrist in the Cartesian space, the objective is to calculate roll, yaw, pitch in the shoulder and also the elbow angle. There are 4 unknown angles to be calculated, however, to make the problem easier, the elbow angle was calculate by having the positions of shoulder, elbow, and wrist in the Cartesian space. This angle is simply defined as the angle between the vector of shoulder-elbow and the vector of elbow-wrist.

Furthermore, the yaw angle can be calculated as the angle between the local Y-axis (Fig. 7.5 (B)) and the normal vector of the limb plane minus  $\pi/2$ . The limb plane is defined as the plane formed by the 3 points shoulder, elbow and wrist. Obviously the normal vector to the plane can be defined as the cross product of the vectors shoulder-elbow and elbow-wrist. Interestingly, the yaw angle is independent of the roll and pitch angles and this means that the yaw angle can always be calculated from the mentioned vectors independent of the roll and pitch rotations. Having calculated these two angles (i.e. elbow, yaw), the inverse kinematic problem minimizes to 2 unknown; roll and pitch which make the calculations significantly easier.

To solve the inverse kinematic problem, we first write the forward kinematics equations, and then try to solve the system of equations. According to the robot arm forward kinematics discussed in chapter 4 (section 4.2.1), the equations of the forelimb are as follows:

$$R \times \begin{pmatrix} 0 \\ L_1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} x_e \\ y_e \\ z_e \\ 1 \end{pmatrix} \quad (7.1)$$

Where in Eq. 7.1,  $L_1$  is the shoulder-elbow arm length,  $(x_e, y_e, z_e)$  is the elbow position and R is the rotation matrix of shoulder:

$$R = \begin{pmatrix} C_r C_y & S_p S_r - C_p C_r S_y & C_r S_p S_y + C_p S_r \\ S_y & C_p C_y & -C_y S_p \\ -C_y S_r & C_r S_p + C_p S_r S_y & C_r C_p - S_p S_r S_y \end{pmatrix} \quad (7.2)$$

In Eq. 7.2,  $C, S$  and  $r, y, p$  stance for  $\cos()$ ,  $\sin()$  and  $roll, yaw, pitch$  angles respectively. Solving the system of equation 7.1 leads to:

## CHAPTER 7. ROBOT'S HARDWARE AND CONTROL

$$\begin{cases} C_p = (\frac{y_e}{L_1 C_y}) \\ S_r = (\frac{ax_e + bz_e}{a^2 + b^2}) \\ C_r = (\frac{-bx_e + az_e}{a^2 + b^2}) \end{cases} \quad (7.3)$$

Where in Eq. 7.3,  $a = L_1 Sp$  and  $b = L_1 CpSy$ .

Eq. 7.3 finally solves the inverse kinematic problem of the forelimb below in that we already know the yaw angle:

$$\begin{cases} p = \cos^{-1}(\frac{y_e}{L_1 C_y}) \\ r = \tan^{-1}(\frac{ax_e + bz_e}{-bx_e + az_e}) \end{cases} \quad (7.4)$$

To calculate the angles of the hindlimb (yaw, pitch and knee) one can again use the inverse kinematic problem; given the ankle position in Cartesian space, and considering two degrees of freedom for the hip model (yaw and pitch, without roll), the objective is to find the yaw, pitch and knee angles. According to the mechanical design of the robot, we used the sequence of pitch-yaw for the hip. In order to make the equations easier, we calculate the knee angle simply by triangular rules when considering the limb as a triangle with known lengths. Similar to the forelimbs, the inverse kinematic here can be solved by deriving the forward kinematic system of equation and try to solve it. The forward kinematics of the hindlimb results in the following transformation matrix (Eq 7.5).

$$T = \begin{pmatrix} C_y & -S_y C_k & -S_y S_k & -S_y(L_1 + L_2 C_k) \\ C_p S_y & C_p C_y C_k + S_p S_k & S_k C_p C_y - S_p C_k & (L_1 + L_2 C_k) C_y C_p + L_2 S_k S_p \\ S_p S_y & C_k S_p C_y - S_k C_p & S_k S_p C_y + C_k C_p & (L_1 + L_2 C_k) S_p C_y - L_2 S_k C_p \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (7.5)$$

Where in Eq 7.5,  $T$  is the transformation matrix composed of rotation and translation,  $C, S$  and  $y, p, k$  stand for  $\cos()$ ,  $\sin()$  and  $yaw, pitch, knee$  angles respectively.

The forward kinematics equations of the limb is given by Eq. 7.6.

$$T \times \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} x_e \\ y_e \\ z_e \\ 1 \end{pmatrix} \quad (7.6)$$

In Eq. 7.6  $(x_e, y_e, z_e)$ , is the position of the ankle. Solving this equation gives the intended angles as:

### 7.3. ROBOT CONTROL

$$\begin{cases} S_y &= \left(\frac{x_e}{L_1 + L_2 C_k}\right) \\ S_p &= \left(\frac{by_e + az_e}{a^2 + b^2}\right) \\ C_p &= \left(\frac{ay_e - bz_e}{a^2 + b^2}\right) \end{cases} \quad (7.7)$$

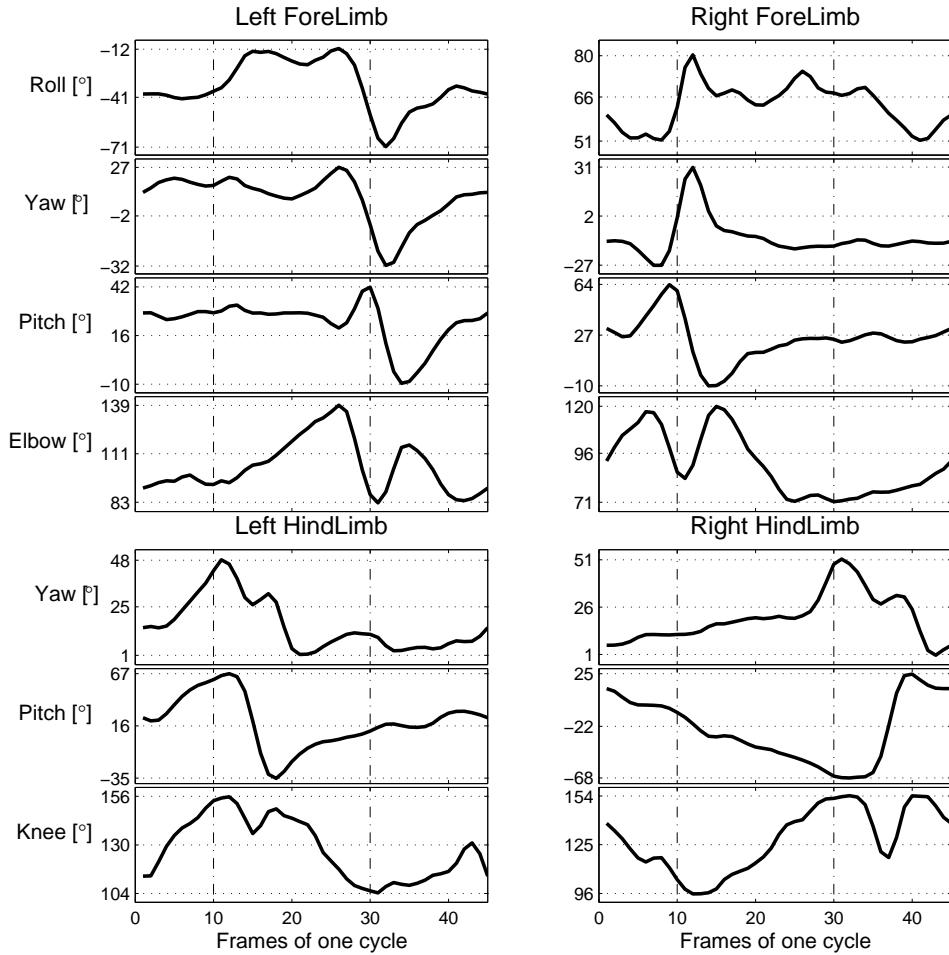
Eq. 7.7 solves the inverse kinematic problem:

$$\begin{cases} y &= \sin^{-1}\left(\frac{-x_e}{L_1 + L_2 C_k}\right) \\ p &= \tan^{-1}\left(\frac{by_e + az_e}{ax_e - bz_e}\right) \end{cases} \quad (7.8)$$

In Eq. 7.8,  $a = C_y(L_1 + L_2 C_k)$  and  $b = L_2 S_k$ .

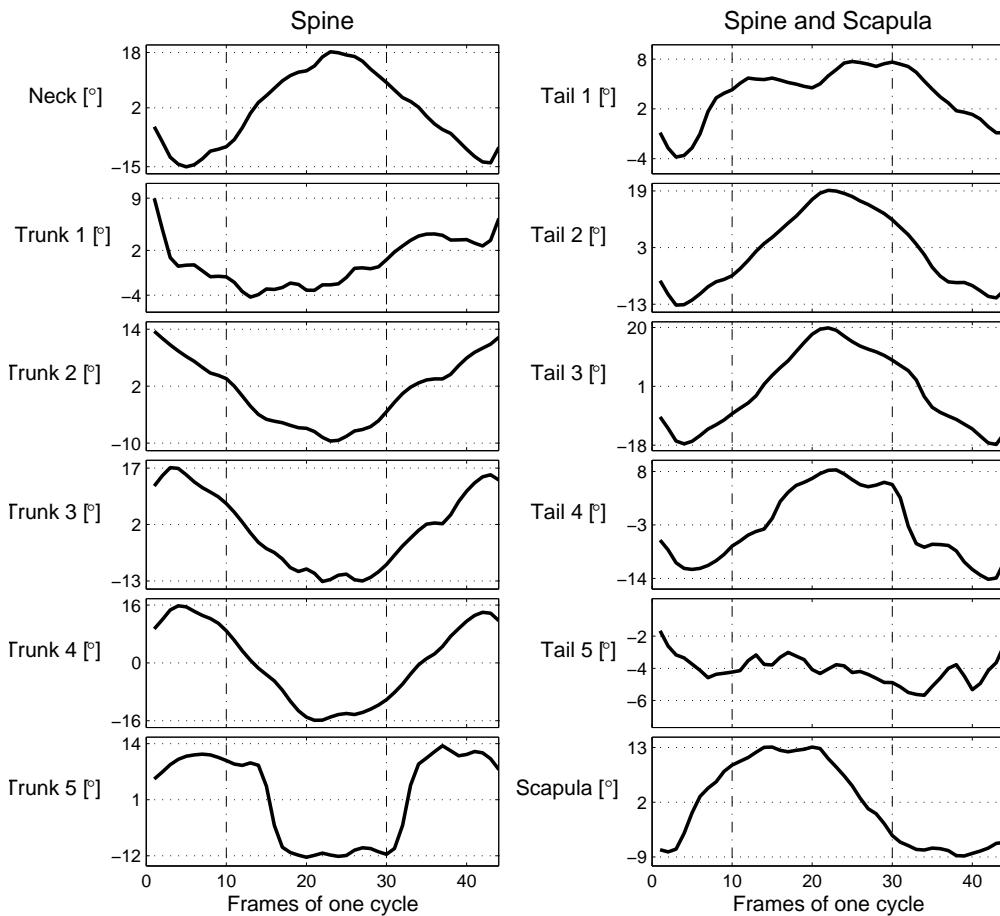
Fig. 7.6 and Fig. 7.7 illustrate an example of the lookup table for a walking gait used to control the robot. Fig. 7.6 shows the angles limbs and Fig. 7.7 indicates the angles of the spine and scapula in one cycle of locomotion. These angles are used for the robot control and they are stored in the micro-controller. As mentioned already, the angles are smoothed and sent to the motors to replicate the kinematics of the animal.

We tested the kinematics on the robot, and the results shown that the robot could perfectly walk. Also it was possible to increase the frequency of the gait, which also indicated robust walking of the robot. We also tested the switching between the gaits, where the robot first walked for some cycles and then switched to swimming posture and started to swim. Though the swimming was done on the ground, the experiment suggested that it was easy to switch between the gaits.



**Figure 7.6.** Angles of the limbs in the walking gait of the salamander in one cycle of locomotion. Each forelimb contains 4 angles; roll, yaw and pitch in the shoulder and elbow angle. Each hindlimbs employs 3 angles; yaw and pitch in the hip and knee angle. All angles are in degree and the horizontal axis represents the frames of a cycle.

### 7.3. ROBOT CONTROL



**Figure 7.7.** Angles of the spine and also the scapula in the walking gait of the salamander in one cycle of locomotion. There is one angle in the neck, 10 angles in the spine; 5 in the trunk and 5 in the tail. The angle of the scapula is also illustrated. All angles are in degree and the horizontal axis represents the frames of a cycle.



# Chapter 8

## Discussion

In this project a methodology was presented in which a salamander-like robot was designed based on the kinematic data collected from a real salamander. The project was started with X-ray kinematic data collection of the salamander by manual tracking of the joints and bones. The collected data was then imported to the developed software for analysis and also to extract any necessary information. A model for the robot was created to represent the real animal. This model employed an optimal number and positions of joints in the spine. Moreover the hindlimbs model were simplified while respecting the capabilities of the robot model to replicate the necessary tasks. After this stage, the robot was designed mechanically in such a way to meet the necessary requirements of the locomotion and to replicate different gaits that the animal employs. Physical simulation of the model verified the correctness of the modeling. Finally at the last stage of the project, the robot was constructed and controlled with the kinematics of the real animal.

The kinematic data collection was subjected to tracking errors. This was due to the fact that the tracking was performed manually although a great effort was made to reduce the error. Also the tracking took a considerable amount of time. This technique can have been done differently for example by developing image processing tools to automatically detect and track the relevant bones. Although the way that we collected the data had unavoidable errors of tracking, it contributed to an accurate method of data collection by tracking the skeleton compared to other techniques that put markers on the body.

Finding a good gait of locomotion for the robot sake was a challenge. A good gait can be considered as a symmetric gait with a repeatable cycle of locomotion. Although the gaits of the animal is supposed to be symmetric, the majority of walking data were not appropriate to be implemented in the robot since they did not employ a repeatable cycle and the cycle change was abrupt. However, the walking gait that was selected for the robot can be categorized as a good one, and the cycle was smoothed to have a regular cycle change.

Another challenge of the project was the foot design of the hindlimbs. Since we used a passive joint for the foot, in general it was hard to control it as we wanted.

## CHAPTER 8. DISCUSSION

The animal foot employs a complex behavior, however with the passive foot, it was really hard to replicate the same movements. Though we reached an acceptable behavior of the foot by springs, the exact replication was something that needed further evaluations, design and time.

As discussed in the previous chapters, a graphical user interface was developed to facilitate the use of the data. Different collected data were imported to the interface and any necessary data were exported out of it. This handy tool indeed made the debugging, testing and verification extremely easy and was used during the entire project. The interface which was developed based on the requirements of the project, was very interactive and user friendly and contributed to make an effective use of the collected data. It also enabled the user to easily select any data (walking, swimming gait, etc) and automatically generate the appropriate data for controlling the robot and simulation. Another important approach of the project was the optimization. The results suggested an optimal use of the spine joints that fitted to different gaits of locomotion such as walking and swimming. In addition the simplification of the hindlimb model contributed to have a simpler model that was easier to implement while respecting the kinematics. By this simplification one degree of freedom from a limb was deleted which can result in less controllability. However, there were distinct advantages such as easier mechanical integration, more suitable mass distribution, center of mass position (compared to the animal), and finally less power consumption. In general the robot employed several degrees of freedom which made it a great tool that can perform many tasks depending on the control strategies. The robot mechanic was robust, reliable, easy to assemble and the robot hardware made it easy to control the robot.

## Chapter 9

# Conclusion Remarks and Future Works

In this project we followed a methodology to design and develop an amphibious salamander-like robot. We obtained the 3D kinematic data of the skeleton of a real salamander from the recorded X-ray videos of ground walking. These data were extracted by manual tracking of the bones and joints that were involved in the locomotion. In addition we developed a comprehensive graphical user interface to facilitate the kinematic analysis. Furthermore by solving the optimization problem for the spine, we determined the optimal number of joints for the robot spine and also their best positions so that the robot can replicate the animal's kinematics. The created model of the robot was verified by simulation. The simulation results showed that the optimization and model simplifications were valid and the model can be implemented in practice. The robot was designed in SOLIDWORKS, the parts were printed in plastic and the entire robot was assembled in a short period of time. Finally the hardware of the robot were programmed and the kinematic data which was determined from the real animal were implemented in the robot. The results suggested that the robot was appropriately capable of walking.

As future work, there are a lot of developments that can be implemented that are outlined as follows:

- The project aimed to design an amphibious robot that could both walk on the ground and swim in water. However in terms of robot construction, it was only expected to focus on the walking part. Obviously the first future work that comes to mind is to further develop the robot so that it can also swim. The waterproofing can be achieved by covering the body with appropriate material while considering the buoyancy rules.
- The robot locomotion is performed by reading the real kinematic of the walking gait of the animal with specific characteristics (e.g. speed, frequency, etc). This means that the robot can walk in a straight line thus far. Hence the robot control can be further developed to perform different gaits (e.g. turning) with different speeds. This can be achieved for example by creating a general model of locomotion based on the kinematics of different gaits of the animal. This

## CHAPTER 9. CONCLUSION REMARKS AND FUTURE WORKS

model should not necessarily be the exact kinematics of the animal but a model that can represent a large range of different gaits. The best approach would be the case that modulating some parameters lead to reach the desired goals.

- Sensory feedback is another work that can be applied to this robot. The most straightforward sensor can be bumper located in the foot/palm to have the events of touching and releasing the ground. Interestingly, the servo motors used in the robot already have several feedbacks such as speed, torque, position, etc, that can be used appropriately. Vision is another useful feedback that can potentially add several benefits to the robot control and perform different tasks.
- Better hardware seems to be inevitable in future works. So far the robot hardware is a programmable micro-controller with restricted memory. The improved hardware will indeed result in better performance of the robot.

# Bibliography

- [1] Hiraoka A. and Kimura H. A development of a salamander robot - design of a coupled neuro-musculoskeletal system. In *Proceedings of the Annual Conference of the Robotics Society of Japan, Osaka*, 2002. (Paper in Japanese).
- [2] Ijspeert A.J. A connectionist central pattern generator for the aquatic and terrestrial gaits of a simulated salamander. *Biological Cybernetics*, 84:331–348, 2001.
- [3] Rivera A.R.V. and Blob R.W. Forelimb kinematics and motor patterns of the slider turtle (*trachemys scripta*) during swimming and walking: Shared and novel strategies for meeting locomotor demands of water and land. *Journal of Experimental Biology*, 213(20):3515–3526, 2010.
- [4] Lundin R.a Ashley-Ross M.A. and Johnson K.L. Kinematics of level terrestrial and underwater walking in the california newt, *taricha torosa*. *Journal of Experimental Zoology Part A: Ecological Genetics and Physiology*, 311(4):240–257, 2009.
- [5] Zahedi K. Hertzberg J. and Pasemann F. Breithaupt R., Dahnke J. Robo-salamander – an approach for the benefit of both robotics and biology. In *Proceedings of the 5th International Conference on Climbing and Walking Robots (CLAWAR 2002)*,, 2002.
- [6] Harischandra N. Cabelguen, J.M. and O. Ekeberg. A 3d musculo-mechanical model of the salamander for the study of different gaits and modes of locomotion. *Frontiers in neurorobotics*, 4:112, 2010.
- [7] Ryczko D. Nagy F. Chevallier S., Ijspeert A.J. and Cabelguen J.M. Organisation of the spinal central pattern generators for locomotion in the salamander: Biology and modelling. *Brain Research Reviews*, 57(1):147–161, 2008.
- [8] Bem T. Delvolve I. and Cabelguen J.M. The evolution of terrestrial locomotion. in major patterns in vertebrate evolution. *Journal of Neurophysiology*, 78(2):638–650, 1997.
- [9] Jenkins Jr. Farish A. and Goslow Jr. The functional anatomy of the shoulder of the savannah monitor lizard (*varanus exanthematicus*). *Journal of Morphology*, 175(2):195–216, 1983.

## BIBLIOGRAPHY

- [10] Evans F.G. The anatomy and function of the foreleg in salamander locomotion. *Anat Rec*, 95, 1946.
- [11] Schmidt M. Haarhaus D. Fischer M.S., Schilling N. and Witte H. Basic limb kinematics of small therian mammals. *J Exp Biol*, 205(Pt 9):1315–38, 2002.
- [12] Jenkins F.A. Gatesy S.M., Baier D.B. and Dial K.P. Scientific rotoscoping: A morphology-based method of 3-d motion analysis and visualization. *J. Exp. Zool*, 2010.
- [13] Gillis G.B. Anguilliform locomotion in an elongate salamander (siren intermedia): Effects of speed on axial undulatory movements. *Journal of Experimental Biology*, 200(4):767–784, 1997.
- [14] Edwards J.L Hecht M.K, Goody P.C. and Hecht B.M. The evolution of terrestrial locomotion. in major patterns in vertebrate evolution. *New York: Plenum*, (1):553–576, 1977.
- [15] M. Hildebrand. *Analysis of tetrapod gaits: general considerations and symmetrical gaits.*, pages 203–236. Plenum Press, New York, 1976.
- [16] Crespi A. Ijspeert A.J. and Cabelguen J. M. Simulation and robotics studies of salamander locomotion. applying neurobiological principles to the control of locomotion in robots. *Neuroinformatics*, 3(3):171–196, 2005.
- [17] Ryczko D. Ijspeert A.J., Crespi A. and Cabelguen J.M. From swimming to walking with a salamander robot driven by a spinal cord model. *Science*, 315(5817):1416–1420, 2007.
- [18] Ijspeert A. J. A 3-d biomechanical model of the salamander. *Proceedings of the second international conference on virtual worlds*, 84:225–234, 2000.
- [19] Edwards J.L. Two perspectives on the evolution of the tetrapod limb. *Integrative and Comparative Biology*, 29(1):235–254, 1989.
- [20] Karakasiliotis K. and Ijspeert A.J. Analysis of the terrestrial locomotion of a salamander robot. pages 5015–5020, 2009.
- [21] Gao K.Q. and Shubin N.H. Late jurassic salamanders from northern china. *Nature*, 410(6828):574–577, 2001.
- [22] Frolich L.M. and Biewener A.A. Kinematic and electromyographic analysis of the functional-role of the body axis during terrestrial and aquatic locomotion in the salamander ambystoma-tigrinum. *journal of experimental biology*, 162:107–130, JAN 1992.
- [23] Ashley-Ross M.A. Hindlimb kinematics during terrestrial locomotion in a salamander (dicamptodon tenebrosus). *Journal of Experimental Biology*, 193(3):255–283, 1994.

- [24] Ashley-Ross M.A. and Bechtel B.F. Kinematics of the transition between aquatic and terrestrial locomotion in the newt taricha torosa. *Journal of Experimental Biology*, 207(3):461–474, 2004.
- [25] Ashley-Ross M.A. and Lauder G.V. Motor patterns and kinematics during backward walking in the pacific giant salamander: Evidence for novel motor output. *Journal of Neurophysiology*, 78(6):3047–3060, 1997.
- [26] D. Ritter. Axial muscle function during lizard locomotion. *Journal of Experimental Biology*, 199(11):2499–2510, 1996.
- [27] Deban S.M. and Schilling N. Activity of trunk muscles during aquatic and terrestrial locomotion in ambystoma maculatum. *Journal of Experimental Biology*, 212(18):2949–59, 2009.
- [28] Reilly S.M. and Delancey M.J. Sprawling locomotion in the lizard sceloporus clarkii: Quantitative kinematics of a walking trot. *Journal of Experimental Biology*, 200(4):753–765, 1997.

TRITA-CSC-E 2011:136  
ISRN-KTH/CSC/E--11/136-SE  
ISSN-1653-5715