

1-1-2011

# Rapid GAIT adoption for snake-like locomotion (RGASL)

Nhan Trung Tran  
*Ryerson University*

Follow this and additional works at: <http://digitalcommons.ryerson.ca/dissertations>

 Part of the [Artificial Intelligence and Robotics Commons](#)

---

## Recommended Citation

Tran, Nhan Trung, "Rapid GAIT adoption for snake-like locomotion (RGASL)" (2011). *Theses and dissertations*. Paper 731.

This Thesis is brought to you for free and open access by Digital Commons @ Ryerson. It has been accepted for inclusion in Theses and dissertations by an authorized administrator of Digital Commons @ Ryerson. For more information, please contact [bcameron@ryerson.ca](mailto:bcameron@ryerson.ca).

# **RAPID GAIT ADOPTION FOR SNAKE-LIKE LOCOMOTION (RGASL)**

by

**Nhan Trung Tran**

B.Sc., Computer Science, Ryerson University 2006

A thesis

presented to Ryerson University

in partial fulfilment of requirements

of the degree of Master of Science

in the Program of

Computer Science

Toronto, Ontario, Canada, 2011

© Nhan Trung Tran 2011



I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.



# **RAPID GAIT ADOPTION FOR SNAKE-LIKE LOCOMOTION (RGASL)**

Nhan Trung Tran

Master of Science, Computer Science, Ryerson University 2011

## **Abstract**

Snake-like robots have low centre of gravity because they are limbless, they also have slender bodies composed of multiple actuating segments. Because of these features, snake robots are widely considered to be most adaptable among all land-based mobile robots. The multi-segmented body that provides their defining characteristic, adaptivity, also brings about the quandary of controlling many actuating segments simultaneously to create directed locomotion. Various methods for snake robot locomotion have been proposed for relatively smooth and flat surfaces. Currently there is no snake robot designed or locomotion method capable of resolving the directed mobility problem in situations where the snake robot is stuck at an impasse, or when it encounters disjointed terrains. There is no method to rapidly create new locomotion that addresses the problem of extensive time delay. This thesis makes the contribution of a modular snake robot called Striker and an elegant solution to create new snake-like robot locomotion on-the-fly, called the Explicit Gait Training (EGT) method. The EGT method allows trainer(s) to rapidly train new kinds of locomotion to address any situation at hand using their knowledge, experiences or even trial and error. The third contribution is the Standard Mobility for Snake Robots (SMMSR) is proposed as a standard platform to evaluate the effectiveness of snake robot locomotion.



## **Acknowledgements**

I would like to express my sincere gratitude to my thesis supervisor, Dr. Alex Ferworn for all his support, guidance and patience. Dr. Ferworn taught me to be agile when the water is at high tide.

I am thankful to Dr. Alireza Sadeghian for his support of this thesis and invaluable advice.

I would like to thank Steve Kanellis, Devin Ostrom, Maria Landau and Gloria Oseiafriyie for their technical support. I have garnered a lot of technical knowledge from these great individuals.

Last but not least, I would like to thank my Mother and Soan Tang who had supported and encouraged me and never once faltered.





# Table of Contents

## Chapter 1: Introduction

1.1	Land-Base Mobility.....	1
1.2	Urban Search and Rescue.....	2-3
1.3	Biological Inspiration.....	3-4
1.4	The Problem.....	4-6
1.5	Contributions.....	6-7
1.6	Thesis Organization.....	8

## Chapter 2: Literature Review

2.1	Introduction.....	9
2.2	Important of Urban Search and Rescue.....	9-10
2.3	Disaster Site Topography.....	11-14
2.4	Biological Inspiration.....	14-16
2.5	Snake Locomotion.....	16
2.5.1	Lateral Undulation.....	16-17
2.5.2	Concertina Locomotion.....	17-19
2.5.3	Sidewinding Locomotion.....	19-20
2.6	Mechanical Design and Mathematical Modeling.....	20-21
2.6.1	Kinematics.....	21-23
2.6.2	Backbone Curve.....	23-24
2.6.3	Serpenoid Curve and Active Cord Mechanisms.....	24-26
2.7	Snake Robots.....	26
2.7.1	Wheeled Snake Robots.....	27

2.7.2	Wheel-less Snake Robots.....	27-28
2.8	Overcoming Friction.....	28
2.8.1	Physical Designs with Passive Casters.....	28-30
2.8.2	Algorithmic Control Locomotion.....	30-31
2.8.3	Simple Sensor-based Locomotion.....	31-34
2.9	Locomotion Control of snake-like robots.....	34-36
2.9.1	Serpentine gait.....	36
2.9.2	Concertina gait.....	36-37
2.9.3	Sidewinding gait.....	37
2.9.4	Lateral Rolling.....	38
2.10	Summary and analysis.....	38-39

### Chapter 3: Striker and Gait Training Method

3.1	Introduction.....	41
3.2	Striker's Metal Skin.....	41-43
3.3	Striker the snake robot.....	43-44
3.4	Striker's communication Architecture.....	44-46
3.5	NSR Micro-controller.....	46-52
3.6	Smart actuator .....	52-55
3.7	Explicit Gait Training Methodology .....	55-58
3.8	EGT Algorithm.....	58-60
3.9	Standard Mobility Metric for Snake Robots .....	60-61
3.9.1	Planar Mobility Metric.....	61-62
3.9.2	Incline Mobility Test.....	62-63
3.9.3	Narrow Passage Mobility Test.....	63-64

3.9.4	Obstacle Mobility Test.....	64-65
3.9.5	Steps Mobility Test.....	65-66
3.10	Summary.....	66-67
Chapter 4: Experiments and Results		
4.1:	Introduction.....	69
4.2	Experimental results	
4.2.1	Experimental results for PMT.....	70-72
4.2.2	Experimental results for IMT.....	72-73
4.2.3	Experimental results for NPMT.....	73-74
4.2.4	Experimental results for OMT.....	74-76
4.2.5	Experimental results for SMT.....	76-79
4.3	The Challenge.....	79-83
4.4	Analysis and Summary.....	83-86
4.5	Summary.....	87
Chapter 5: Conclusion and Future Work		
5.1	Conclusion.....	89-91
5.2	Future Work.....	92
Appendix A.....		
Appendix A.....		93-94
Appendix B.....		95-110
Appendix C.....		111
Appendix D.....		112-113
Bibliography.....		114-123

## List of Tables

Table 3.5: Specification of NSR microcontroller for Striker.....	49
Table 4.23: SG1 gait created with the EGT method.....	74
Table 4.2.4: The SG2 gait has 20 poses .....	75
Table 4.2.5: SG3 joins angles of 24 Frames.....	77
Table 4.3: The Hillz Gait is compose of 24 FRAMES.....	81-82
Table 4.4: Experimental results.....	85-86

## List of Figures

Figure 2.3.1: Vertical stacked collapsed.....	12
Figure 2.3.2: Collapsed building.....	13
Figure 2.3.3: Gaps and void Created.....	13
Figure 2.4.1: Python snake skin and skeleton.....	15
Figure 2.4.2: Lateral articulation.....	15
Figure 2.5.1: Lateral Undulation.....	17
Figure 2.5.2A: Lateral Undulation with lateral constrains.....	18
Figure 2.5.2B: Lateral Undulation without lateral constrain.....	19
Figure 2.5.3: Sidewinding locomotion.....	20
Figure 2.6.1: Denavit-Hartenberg guide to obtaining kinematics parameters.....	22
Figure 2.6.3A: Graphical depiction of Serpenoid curve.....	25
Figure 2.6.3B: Active cord mechanisms III using Serpenoid curve .....	26
Figure 2.8.1A: Snake robot with passive casters.....	29
Figure 2.8.1B: The lateral friction.....	29
Figure 2.8.3: Aiko's segment with suspended protective shell.....	33
Figure 2.9: Model of snake robot joints configuration.....	35
Figure 3.2A: Prototype S1.....	42
Figure 3.2B: Striker's metal skin.....	43
Figure 3.3: Striker; composed of on NSR microcontroller.....	44
Figure 3.4.1: Multi-drop interconnection and data transmission.....	45
Figure 3.4.2: Packet structure.....	45
Figure 3.5A: NSR microcontroller with tri-state buffer.....	48
Figure 3.5B: NSR Algorithm.....	52

Figure 3.6.1: Smart actuator position.....	53
Figure 3.6.2: The NSR Microcontroller.....	54
Figure 3.6.3: Smart actuator algorithm.....	55
Figure 3.8: Explicit Gait Training Algorithm.....	59
Figure 3.9.1: Planar Mobility Test.....	62
Figure 3.9.2: Incline Mobility Test.....	63
Figure 3.9.3: Narrow Passage Mobility Test.....	64
Figure 3.9.4: Obstacle Mobility Test.....	65
Figure 3.9.5: Steps Mobility Test.....	66
Figure 4.2.1A: Sidewinding gait in four distinct poses.....	71
Figure 4.2.1B: The lateral rolling gait.....	72
Figure 4.2.2: Striker scaling.....	73
Figure 4.2.3: SC1 gait.....	74
Figure 4.2.4: Striker overcoming OMT.....	76
Figure 4.2.5: Striker climb stair SG3.....	78
Figure 4.3.1: Concrete inclined surface.....	80
Figure 4.3.2: Metal skin augmentation.....	81
Figure 4.3.4: Using the Hillz gait.....	83

## **Procedure:**

Procedure 3.5: Convert radian angle to smart actuator position value.....	51
---	----

## **Equations:**

Equation 1: Homogeneous transformation matrix.....	23
Equation 2: Serpenoid curve.....	24
Equation 3: Sinusoidal curve.....	35
Equation 4: Serpentine movement.....	36
Equation 5: Concertina movement.....	37
Equation 6: Sidewinding gait.....	37
Equation 7: Lateral rolling.....	38





# Chapter 1

## Introduction

### 1.1 Land-Based Mobility

Effective locomotion on diverse and rugged terrain is the defining feature of land-based mobile robots. Land-based mobile robots are increasingly being utilized in many facets of modern societies. Predominantly wheeled and tracked robots are used for urban search and rescue (USAR), military reconnaissance, exploration, maintenance or inspection. Wheeled and tracked robots primarily depend on a form of rotary motion mechanism to drive them. This type of robot is relatively simple to design and operate. Wheeled robots are suited for high speed locomotion on level and smooth continuous terrains.

Wheeled and tracked robots are not ubiquitous, they are not effective in all terrains. Wheeled and tracked robots are not effective with low shear surfaces such as sandy deserts and/or muddy surfaces. Rugged terrains or ones with chaotic, highly variable, topography such as surfaces typically found within disasters. These terrains can drastically reduce a wheeled and or tracked robot's ability to move effectively if not completely strand it. Legged robots are thought to be more adaptable than wheeled robots and are actively being investigated [1-8]. Perhaps the most flexible form of robot locomotion is that exhibited by the biologically inspired snake-like robots with multi-segmented bodies possessing many degrees of freedom (DOF).

## **1.2 Urban Search and Rescue**

Urban search and rescue operations have become very visible in the media. Globally, people migrating to city-centres, a process known as Urbanization [9-10]. Because of urbanization approximately fifty percent of world population are in urban-centres [10-13]. People migrate to urban-centres to seek employment, education, and a better living standard. Urbanization brings about a situation where there is a very large population concentrated in a relatively small geographical space. Thus when disasters, whether natural or manmade, hit urban centres, more people are affected; the high density of the population increases the possibility of human injuries and deaths.

When a disaster occurs in a city area, USAR teams are deployed to the site(s). Upon arriving in the area, USAR teams of first responders are faced with scenes of chaos and destruction of buildings and other structures. First responders conduct a search for survival victims and casualties. After victims are found on the surface of relatively safe areas within a disaster, the search will expand to more challenging areas generally focussing on collapsed buildings, and other damaged structures such as bridges and highways where people are likely to be found. These sites are composed of collapsed building materials: concrete sheets, wreckage, steel beams, rebar and everything else that once formed the structures and now pose threats to both victims and would-be rescuers. These premises are very dangerous due to their instability and may further collapse when disturbed. Topographically, the site terrains are chaotic; therefore, the first responders' preliminary tasks prior to the search are to survey the sites and make them safe before they can enter and conduct the actual search. Preliminary cautionary steps may take

several hours or days to prepare. Instead of human first responders physically conducting the search, rescue robots may be employed. With the use of robots most of the precautionary tasks may be bypassed as the loss of a robot is much more acceptable than the loss of a human life. With the use of rescue robots, first responders are able to conduct searches immediately upon arriving at the sites. The use of robots may also reduce the likelihood of further injuries for first responders and decrease the time it takes to get on to the search site; thus, resulting in more people found faster and potentially faster rescues. Urban disaster survival rates are a direct function of time--the longer victims remain in the rubble the lower their chances of survival. The extremely disturbed nature of typical disaster sites requires rescue robots to be highly adaptable. One type of robot that shows the greatest potential for adaptation is the snake-like robot.

### **1.3 Biological Inspiration**

Snake-like robots are biologically inspired. Snakes are physiologically simple; they are limbless with long tubular bodies with simplistic and repetitious spine joints [14]. Each individual joint has a limited motion but the sum of all their spinal segments allows the entire body to achieve very flexible and complex motions. Snakes possess one of the most versatile forms of locomotion among all creatures. Snakes deform their internal body structures by muscle contractions to create choreographed sequences to move. The versatility of snakes' locomotion allows them to thrive on all continents with the exception of Antarctica. Snakes live in widely diverse terrains: They scale steep rock cliffs, traverse narrow passages, climb trees and navigate through dense bushes.

Furthermore, snakes live in hostile desert environments and navigate on the low shear sand surfaces and even swim in water.

Similar to biological snakes, snake-like robots are also limbless with long tubular bodies composed of many individual segments. Generally, all body segments are modular and very similar or identical to one another. Each segment possess actuator(s) that can articulate with at least two degrees of freedom (DOF), while other snake robots have links with three or more DOF. These modular segments are serially joined from one to the next forming a long chain. The modularity of the segments that make up the body of snake robots result in a robot that is robust and greatly adaptable.

## **1.4 The problem**

The pioneering studies of snake-like robots were conducted by Shigeo Hirose [15] [16]. Hirose studied the movements of snakes and used them as inspiration for his robots. He formulated the serpenoid curves that described the movements of snakes on a constant-friction surface [16]. He demonstrated this principle with a multi segmented snake-like robot called Active Cord Mechanism Model 3(ACM-3). ACM-3 has wheels placed along the sides of its segments and is capable of moving on a planar surface without directly driving its wheels. The serpenoid curve is analogous to a snake slithering forward through muscle contraction. Since ACM-3, many other snake robots<sup>1</sup> have been developed with multiple segments and have very large or infinite DOF.

---

<sup>1</sup> Note: The terms snake-like robot and snake robot are used interchangeably throughout this report.

Snake robots are classified as redundant or hyper-redundant robots[ 17-19]. Kinematic redundant robots are capable of performing tasks in a multitude numbers of diverse ways, while hyper-redundant robots are capable of almost endless ways of performing the same task. The redundancy in snake robots results in their defining characteristic, adaptivity, but also brought about the quandary of controlling their many actuating segments. Snake robots are limited by the types of locomotion that are programmed into them. These movements are referred to as gaits [20].

Snake robot gait controls are often very complicated since snake robots are made up of multiple segments connected by joints which can possess high numbers of DOF. Control algorithms must precisely generate specific joint angles for all the joints at each time step in order to create directed motion. Several techniques have been proposed to create different snake gaits, [15] [18-19] [21] , all of which rely on some form of centre line curve constraints while others depend on physical designs [22-24]. Unlike wheeled and tracked robots, these complex control algorithms have many input parameters that must be specifically tuned based on each individual snake robot for a specific type of surface. The tuning process is arduous and time consuming. The process of creating a new gait control algorithm may take weeks to months when using the current methods.

The problem is further compounded when the terrains are complex. Up until now each snake robot has been tested in well-controlled settings with relatively simple continuous flat terrains. No snake robot has ever been used in real world settings. USAR search grounds, which consist of collapsed buildings, structures and piles of rubble, are

impossible for current snake robots to traverse. USAR search sites present the worst type of terrain one can find; they are composed of many obstacles with sharp edges, highly uneven and unstructured surfaces, crevices, gaps and openings. USAR terrains can be simply summed as disjointed and chaotic. The environments and situations require a new solution to overcome impasses quickly. Thus the question is: How can we create new gaits quickly enough for snake robots to avoid becoming stuck on impasses such as gaps of unknown height, irregular obstacles and cracked surfaces?

## **1.5 Contributions**

This thesis makes contribution in three areas:

1. The Explicit Gait Training method for rapid gait creation.
2. Striker, a modular and tethered snake robot.
3. Standard Mobility Metric for Snake Robots to evaluate the effectiveness of snake robot locomotion.

The first contribution of this thesis is to provide a method to create new gaits for snake robots to address obstacles and topographies that are not preprogramed, called Explicit Gait Training (EGT). We believe the EGT method is an elegant solution to rapidly create new specific gait(s) tailored to any situation utilizing the trainers' expert knowledge of the terrain. The application of the EGT method is intended to overcome a difficult terrain having disjointed and irregular surfaces.

The second contribution is the development of a modular and tethered snake robot called, “Striker” at the N-CART<sup>2</sup> laboratory in the department of Computer Science at Ryerson University. Striker is modularly designed with a body composed of multiple identical segments. Each segment possesses 3-DOF and is encased in a metal bracket.

The third contribution is the proposal of a standardized framework to evaluate and report the effectiveness of snake robot locomotion, called the Standard Mobility Metric for Snake Robots (SMMSR). SMMSR consists of a set of five tests; tests are decompositions of different surfaces commonly found in collapsed buildings and other structures. We propose SMMSR to serve as a standard platform to qualitatively evaluate and quantitatively benchmark different land-based snake robots; it provides a means to bring transparency to reported research results and a basis for comparison among different snake robot abilities. Striker will be the first to be evaluated using SMMSR.

---

<sup>2</sup> N-CART: Network-Centric Applied Research Team



## **1.6 Thesis Organization**

This thesis is organized as follow:

Chapter 2 presents a literature review of urban search and rescue, disaster terrains and topography, biological inspiration, snake robots, and control algorithms.

Chapter 3 discuss the technology and technical specifications of our snake robot Striker, and the Explicit Training Gait method and Standard Mobility Metrics for Snake Robots.

Chapter 4 presents experimental results, and observations and analysis of our work.

Chapter 5 provides conclusions and discusses potential future work.

## **Chapter 2**

# **Literature review**

### **2.1 Introduction**

This chapter presents the importance of USAR and typical terrain found in urban disasters. Secondly, natural snakes and their gaits are discussed as models for the designs of robotic snakes. The chapter goes on to discuss various snake robot developments, both wheeled and wheel-less, along with several control techniques.

### **2.2 Important of Urban Search and Rescue**

In recent years, USAR has become a prevalent issue that pushed the United Nations to create a group called the International Search and Rescue Advisory Group (INSARAG). INSARAG was established in 1991 and currently has over 200 representatives in USAR discipline from more than 80 nations. INSARAG's purposes are to share urban disaster response strategies and define guidelines [25]. Urbanization causes urban centres to become overcrowded [10-11] therefore; when a disaster occurs the impact is more devastating as there are simply more people around who can become involved in it.

Search and rescue at any disaster site can be viewed in three distinct stages. First, gather data by searching the site for surviving victims. Second; formulate and strategize rescue plans based on the data gathered in the search. Lastly, execute the plan to retrieve people

by extricating, medically stabilizing and extricating them to second line medical facilities.

The first stage is the most critical stage because it determines the course of actions that follow. Furthermore, the first stage determines how quickly victims may be rescued or at least have vital resources delivered to them. After arriving at the search site, first responders must survey the site to determine whether there are any hazards including if it is structurally safe to enter an area. Sections that are considered to be structurally dangerous will be avoided during the search and if these sections must be searched, then shoring, cribbing and other structural supports must be put in place before the search can proceed. These preliminary precautionary steps are needed to minimize the likelihood of first responders being injured. The precautionary steps are necessary but they take a lot of valuable time and time is in short supply for people trapped under rubble.

Instead of human first responders, there may be situations when rescue robots such as snakes and other types of robots may be useful for searching areas that cannot be directly accessed by human searchers. With the increasing prevalence of toxic material produced through terrorist activity or simply released by the collapse of a building, it has been predicted that rescue robots may play an ever more important role in dealing with the aftermath of a disaster [26-32].

## **2.3 Disaster Site Topography**

Disaster sites are sure to be chaotic, filled with debris and bereft of working systems. However on closer inspection one can observe patterns. Modern metropolitan areas are primarily constructed of concrete and steel with modern construction techniques; thus, when a disaster strike not everything gets obliterated. Often reinforced concrete ceilings, floors and walls form predictable structures as a result of a structural collapse.

One structural collapse configuration known as a “pancake stack collapse” [33] occurs when the supporting columns or structures of a building are no longer able to support the weight of the floors above. The cascading effect of the top level of the building collapsing straight down onto the lower levels creates a stack of collapsed floors. This type of formation can be found in disasters that involve relatively short buildings or multi-level parking garages and multi-level freeway. In this scenario the flooring of each level stays relatively intact, and there are spaces remaining between the floors, wedged by things the floor holds or by the withstanding support structures (Figure 2.3.1). The space in between the floors may be large enough to permit survivals but the settings are very dangerous and confining.



Figure 2.3.1: Pancake collapse: Collapsed parking at Disaster City USAR Training facility, College Station, Texas, U.S.A.

A configuration that is certain to be present is when the floor or ceiling comes loose and becomes wedged between standing walls and the lower floor or ground forming inclined surfaces as shown in Figure 2.3.2. These surfaces stop dangerous fragments and debris but more importantly they create crevices and voids that may potentially hold survivors. Such voids are shown in Figure 2.3.3.



Figure 2.3.2: Collapsed building with floor sheet wedge between standing structure.



Figure 2.3.3: Gaps and void created by wedged concrete sheets.

The terrain surrounding and inside these gaps and voids are inhospitable often making them impossible to reach by humans and certainly by current mobile robots. Snake

robots with high adaptivity and small cross section have the potential to get through the rubble and gain access to the voids where people may be trapped.

## **2.4 Biological Inspiration**

Biological snakes and their limbless locomotion are the source of inspiration for most limbless mobile robots [15-16] [19]. Thus, it is necessary to start with a brief introduction to the physical characteristics of snakes allowing for a better understanding of the correlation between the biological and mechanical counterpart of these reptiles.

Snakes have the largest number of vertebrae compare to any other animal. Depending on the species, snakes can possess between 100 to 400 vertebrae [34], which allow them to exhibit complex vertebral articulation as shown in Figure 2.4.1. Snakes, like most vertebrates have spines that do not stretch far, as such, they tend to maintain a straight and constant length when moving. Moreover, their skeletal structures are relatively simple in comparison to other vertebrates. Their skeletons consist of only three types of bones: a skull, vertebrae, and ribs.

Each Vertebra is limited to 10-20 degrees of lateral articulation, while dorsal plus ventral articulation is limited to about 10 - 15% of the lateral articulation as shown in Figure 2.4.2.

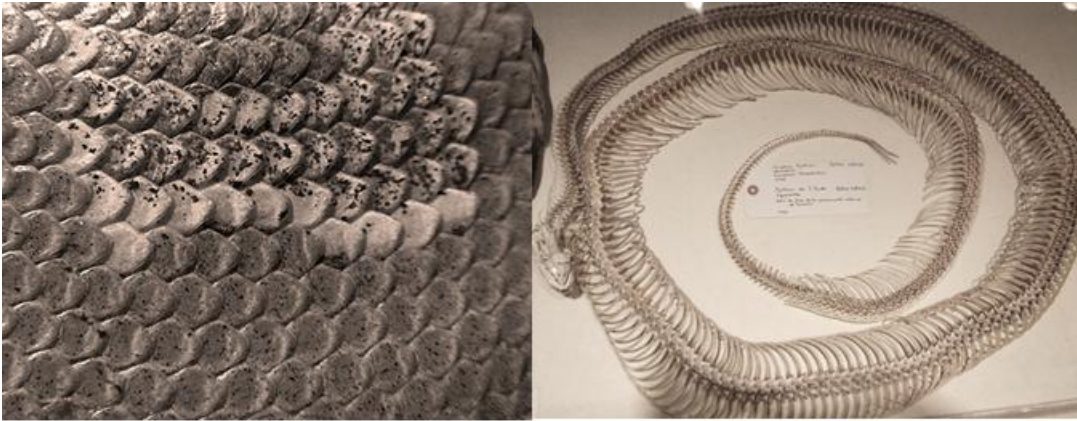


Figure 2.4.1: Python snake skin and skeleton

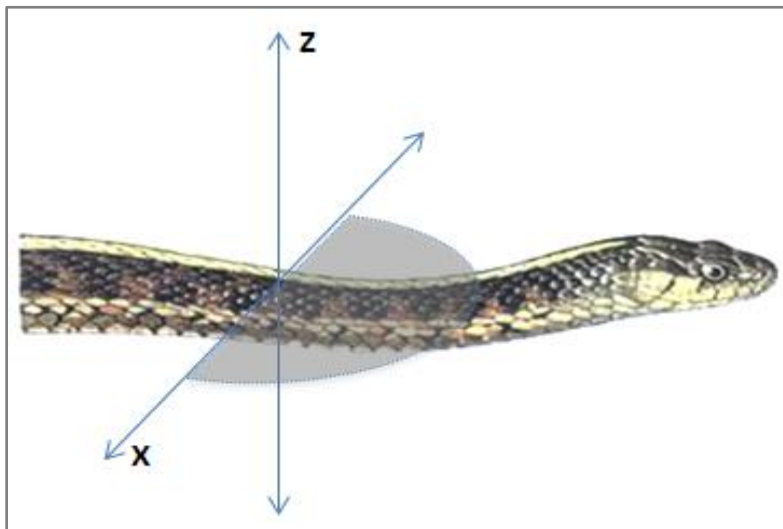


Figure 2.4.2: Lateral articulation is articulation in the X plane. Dorsal and Ventral articulation is articulation in the Z plane.

Remarkably, vertebral physiology also inhibits longitudinal torsion which would otherwise twist the spine. Although each spinal joint has limited motion, the sum of these articulations can result in large angles and complicated shapes [35]



Snakes possess a system of tendons and muscles that interconnect various vertebrae with ribs and also vertebrae with membranes. The skin or the membranes are covered with overlapping scales (Figure 2.4.1). Scales are characteristically hard and smooth providing protection and are used for locomotion [20]. The features of the scales permit snakes to slide with minimal friction in the intended direction. In one study, Gray and Moon showed that the coefficient of friction of grass snake on various substrates is very small [35][20].

## **2.5 Snake Locomotion**

The versatility of snake locomotion allows snakes to thrive in widely diverse terrains. They scale steep rock cliffs, traverse narrow passages, climb trees, navigate through dense bushes and vegetation with very few types of locomotion. The most commonly used locomotion employed by snakes are, lateral undulation, concertina locomotion, and sidewinding locomotion.

### **2.5.1 Lateral Undulation**

Lateral undulation otherwise known as serpentine locomotion is the most commonly used gait by snakes. It is a continuous movement of the entire body in a wave-like pattern similar to that of a sinusoidal curve [20]. The snake's underbelly is in constant contact with the surface over which the snake travels as shown in Figure 2.5.1. The waves propagate laterally from head to tail pushing against the ground surface and generating forward movement. During locomotion, each point of the body follows the same path established by the head and neck resulting in directional steering.

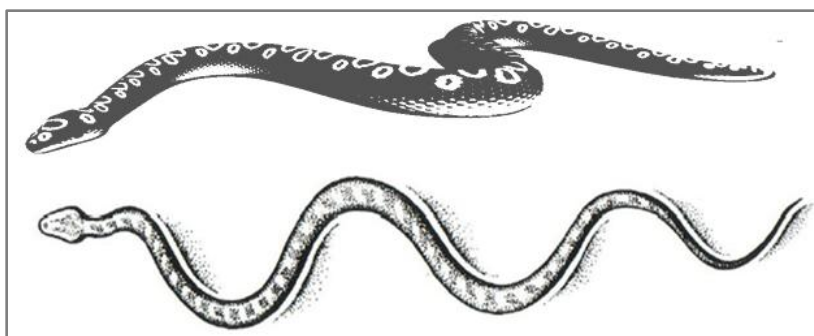


Figure 2.5.1: Lateral Undulation view from side and top view perspective.

One factor affecting efficiency of lateral undulation is the small body curvature relative to body length [34]. Such factor often restricts this gait to smaller snakes. Snakes such as constrictors have a larger body diameter to body length ratio; thus, they require higher numbers of vertebrae to form the curvature for effective undulation. This is not applicable to snake robots because of their wide pivoting joints. The fastest species of snakes can attain speed up to 11km/hr. because their body lengths are approximately 10 to 13 times longer than their body circumference [35].

## 2.5.2 Concertina Locomotion

The concertina gait derives its name from the small accordion-like instrument which is similar in shape to the snake's body when it is in motion. Forward movement is generated via a series of alternating pulling and pushing of the body into relatively small sets of waves at either end of the body. Forward motion can be generated by the higher coefficient of friction in the folded section relative to the rest of the body. Concertina gait can be strategically applied in two types of scenarios. Primarily, concertina is effectively used in narrow passages, pipes, or environments where lateral restraints are available.

The movement algorithm is as follows: Given that the section near the head is folded, the folds anchor the body to the ground and laterally against the side, thus allowing the posterior section to be pulled forward. The front is then extended forward and before the end is completely unfolded, the front section begins to fold, thus completing the sequence of motion (Figure 2.5.2A).

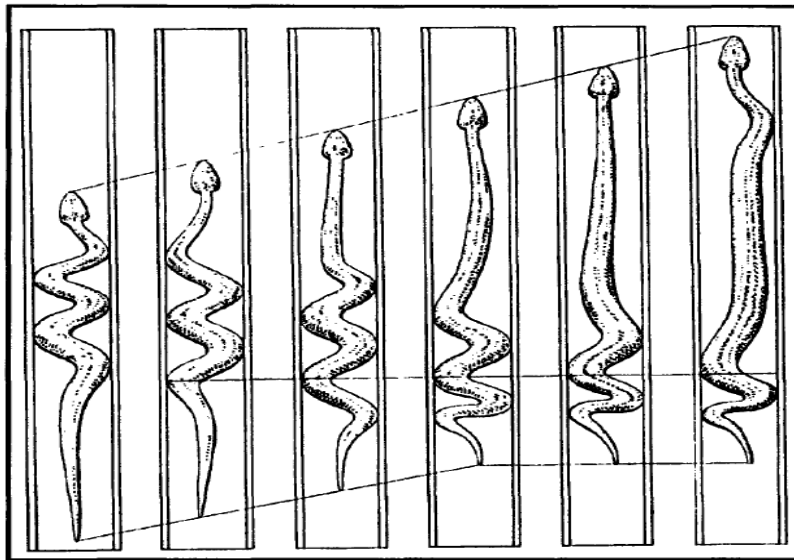


Figure 2.5.2A: Lateral Undulation with lateral constraints through a narrow passage.

Secondary, Concertina movement strategy is also used when a snake is moving along branches or in situations where lateral surfaces are not available. In this gait variation the snake's body is folded at the midsection. The folded sections and the tail are anchored to the surface while the front is extended until the entire body is straightened. The frontal section of the snake then becomes anchored while pulling the hind section forward to create several midsection folds (Figure 2.5.2B). Snakes also use their ventral scales found along the side of their bodies to create additional traction for this type of locomotion.

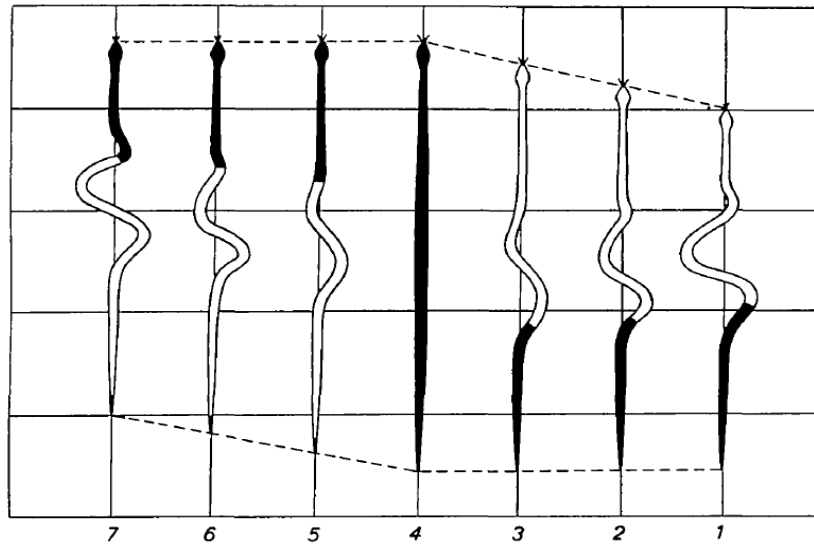


Figure 2.5.2B: Lateral Undulation without lateral constrain.

### 2.5.3 Sidewinding Locomotion

Sidewinding Locomotion is considered one of the more remarkable gaits. On first inspection, this locomotion seems counter intuitive because it involves movement on slippery surfaces or surfaces with low shear while maintaining only two to three contact points. The snake lifts and curves its body in an upward and forward position and then lays down again in a rolling movement in the direction of motion as depicted in Figure 2.5.3. The motion is periodic with each body segment performing actions of those segments before it. As with other gaits, the placement of the head angle determines directionality. Direction changes and periodic motion changes originate from the head and propagate down to the rest of the body [34]

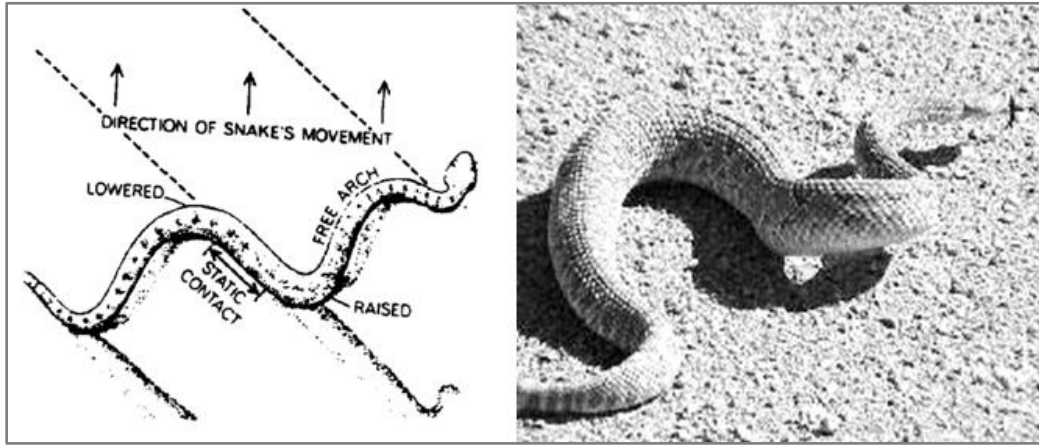


Figure 2.5.3: Sidewinding locomotion

In contrast to lateral undulation, sidewinding employs brief static contacts with the surface. The snake generally establishes three contact points with the ground, but may also have only two as observed by Burdick [36]. The body weight is distributed among these points; thus, maximizing frictional force at those locations. Hirose describes the basic motion dynamics as the approximation of two sine waves propagating along the body. One sine wave describes the horizontal plane and the other describes the vertical plane. The gait is controlled by varying the amplitude, phase offset, and frequency of these two sine waves [37].

## 2.6 Mechanical Design and Mathematical Modeling

Mathematical models of snake-like robots are completely dependent on their mechanical designs. Basic properties of mechanical designs include the type of joints, number of DOF and the presence of casters (wheels). Most snake-like robot's links are connected

together with joints that are either revolute or prismatic joints, while the former is more common than the latter. A revolute joint is like a hinge that has relative rotational motion along the axis and a prismatic joint allows for relative linear motion between two joints. However, there are specialized joints presented [22] [38] [39], specifically designed to achieve 3-D articulations per joint. Increased DOF drastically increase computational requirements and control complexity, but most snake robot controls are based on well-defined curves and or sinusoidal waves which reduces control complexity.

### **2.6.1 Kinematics**

The goal of kinematics is the mapping of joint space with Cartesian task space. This is done by computing the position and orientation of the end-effector as a function of joint variables. The system is characterized by  $N$  degrees of mobility or DOF, and each DOF is associated with a joint articulation also known as the joint variable ( $q$ ). As a result, given the angle of rotation for each joint angle, we can compute the geometrical aspect of the motion [40].

The Direct kinematics is the mapping of joint space to Cartesian space. This means that the end-effector position and orientation is obtained through joint variables. Kinematics is recursive and is obtained systematically with simple products of the homogeneous matrices with respect to a reference frame. Variables needed to obtain direct kinematics are  $a_i$  is the distance between  $O_i$  and  $O_{i+1}$ ,  $d_i$  coordinate of  $O_{i+1}$  along  $z_{i+1}$ . Where  $\alpha_i$  is the angle between axes  $z_{i-1}$  and  $z_i$  about axis  $x_i$  to be taken positive when rotation is made

counter-clockwise,  $\theta_i$  is the angle between axes  $x_{i-1}$  and  $x_i$  about axis  $z_{i-1}$  to be taken positive when rotation is made counter-clockwise as shown in Figure 2.6.1.

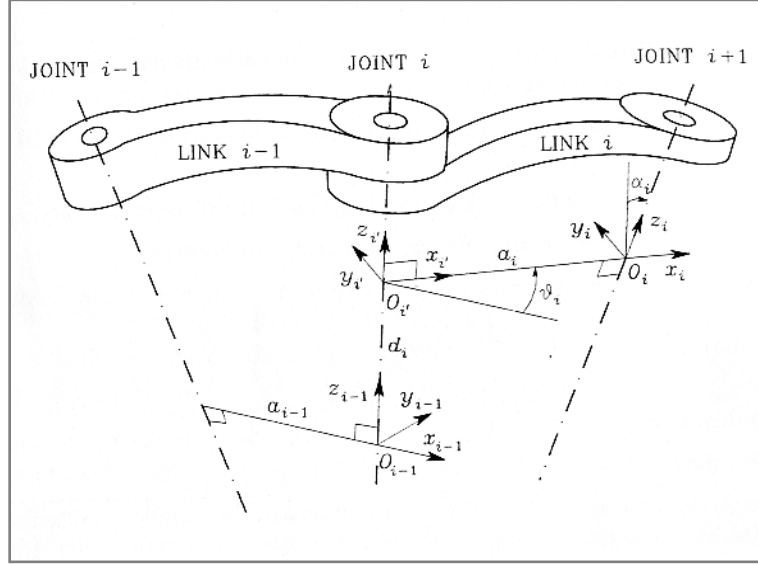


Figure 2.6.1: Denavit-Hartenberg guide to obtaining kinematics parameters [40]

Two of the four parameters ( $a_i$  and  $\alpha_i$ ) are always constant and depend only on the geometry of connection between consecutive joints. The only one ( $\theta_i$ ) of the last two variables are required because the joints used in this research are only revolute. Once the Denavit-Hartenberg (D-H) parameters have been obtained, it is possible to express the coordinate transformation between Frame  $i$  which connects Link  $i-1$  to Link  $i$  with the homogeneous transformation matrix,  $A_i^{i-1}(q_i)$ , [41] presented by equation (1).

$$\mathbf{A}_i^{i-1}(q_i) = \mathbf{A}_{i'}^{i-1} \mathbf{A}_i^{i'} = \begin{bmatrix} c_{\vartheta_i} & -s_{\vartheta_i} c_{\alpha_i} & s_{\vartheta_i} s_{\alpha_i} & a_i c_{\vartheta_i} \\ s_{\vartheta_i} & c_{\vartheta_i} c_{\alpha_i} & -c_{\vartheta_i} s_{\alpha_i} & a_i s_{\vartheta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

*Note:  $c_\theta$  and  $s_\theta$  are short form for cosine theta and sin theta, respectively*

D-H convention is usually used to describe robots with fixed base frame, thus a modification of this convention is adopted to compensate for non-fixed base snake-like robot is described in [42][43].

Reference [42] presents a snake robot consisting of five segments of three DOF each. Each module is parameterized with D-H convention. Additionally, a modification is made by placing the base coordinate system on the motionless link: which is commonly the head or tail link. This snake robot actuates only four of five modules simultaneously. Consequently, the position and orientation is obtained relative to the first motionless link. Locomotion algorithm described in [42][43] allows for the movement of all joints but introduces the use of a virtual structure for orientation and position (VSOP) which allows for description of the kinematics of the snake robot with respect to the inertial reference frame.

### 2.6.2 Backbone Curve

Unlike the Denavit-Hartenberg convention which begins by obtaining the position and orientation of each joint directly, the Backbone curve describes the shape of the spine of the snake-like robot. In this method, a backbone curve is defined as a piecewise continuous curve that captures the overall geometric features of the snake-like robot [43].



The backbone curve is restricted by a set of shape functions which models the form. The model is undefined when the shape is straightened. A set of orthonormal reference frames are found along-side the backbone curve to specify the actual snake robot configuration. The problem arises when one is trying to solve the inverse kinematics problem to obtain the joint angles given the end point position. For snake-like robots and snake-like arms, this is computationally expensive. The problem must be simplified in order to apply the backbone curve in determining the proper time variance behaviour [43-45]. This method is very well suited for abstracting, understanding and developing the shapes for snake-like robot with motion planning. In addition, this method might not necessarily pertain to mobile application due to its computational complexity.

### 2.6.3 Serpenoid Curve and Active Cord Mechanisms

The first qualitative research on snake locomotion was conducted by J. Gray in 1946 [20], but the pioneer builder of snake-like robot is Shigeo who studied the movement of living snakes and used their inspiration for his robots. Shigeo photographed the movement of snakes and concluded that serpentine locomotion followed a curve called “serpenoid” curve, which has curvature that sinusoidally varies along the longitudinal axis of the snake’s spine [37]. The formula for the serpenoid curve is as follows:

$$\begin{aligned}
 x(s) &= sJ_0(\alpha) + \frac{4l}{\pi} \sum_{m=1}^{\infty} \frac{(-1)^m}{2m} J_{2m}(\alpha) \sin\left(m\pi \frac{s}{l}\right) \\
 y(s) &= \frac{4l}{\pi} \sum_{m=1}^{\infty} (-1)^{m-1} \frac{J_{2m-1}(\alpha)}{2m-1} \sin\left(\frac{2m-1}{2}\pi \frac{s}{l}\right)
 \end{aligned} \tag{2}$$

Where  $x(s)$ ,  $y(s)$  are displacement in  $x$ ,  $y$  direction respectively,  $S$  curve body length,  $l$  is the actual links length (where is assume to be equal),  $J_*(.)$  = Bessel function, and  $\alpha$  is the initial winding angle. Depiction of the Serpenoid Curve is shown in Figure 2.6.3A.

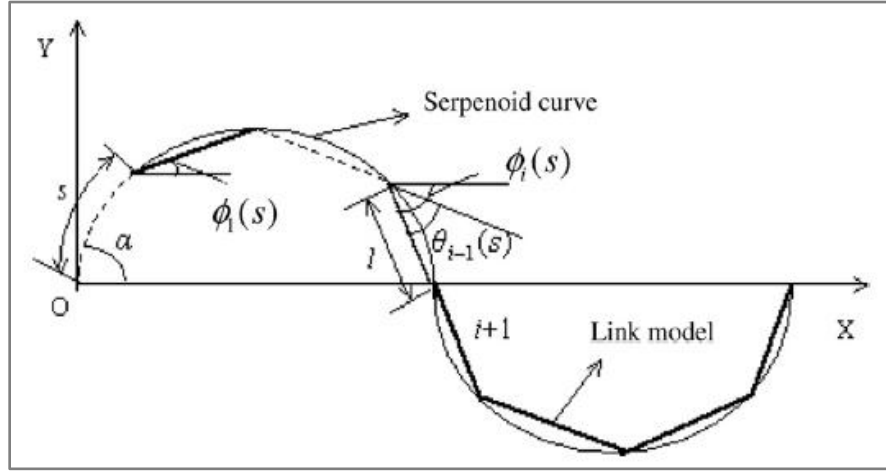


Figure 2.6.3A: Graphical depiction of Serpenoid curve.

He later realized that movement of the snake on a surface with constant friction can be approximated by the serpenoid curve equation. Among the first snake-like robots is a planar multi-segmented wheeled vehicle that follows the serpenoid curve which generates forward movement when internal torque is applied to its segment. This movement is known as the active cord mechanisms (ACMs), analogous to that of muscle contractions in snakes [15]. Wheels on the robot are passive casters that are used primarily to prevent lateral movement of its segments, and to allow for constant friction which is needed for the serpenoid curve, as shown in Figure 2.6.3B.



Figure 2.6.3B: Active cord mechanisms III using Serpenoid Curve

An alternative way of applying ACMs was demonstrated in [19], where the first five joint angles are controlled inputs and the rest of the posterior joint angles are inherited from the first five. ACMs provide a simple method for producing snake locomotion without resorting to complex numerical analysis found in the backbone curve method.

## 2.7 Snake Robots

Ever since the first mechanical application of lateral undulation by Hirose [15], there have been a steady number of snake robots developed. A survey of some of these robots will be discussed in the next section. Furthermore, snake-like robots are explicitly separated into wheeled and wheel-less categories, while implicitly separating them into planar and 3-axial articulations.

### **2.7.1 Wheeled Snake robots:**

Wheeled robots utilize passive casters that create variations of lateral undulation. This method can be found in the work of Hirose and Umethani [15][16], Ostriwski and Burdick [19] Gavin Miller [46], ye et al [47]. Due to the planar nature of robots with passive casters, the only other locomotion that is achievable is lateral rolling. This is a less common type of gait where side propulsion is generated by the periodic movement into a parabolic curve followed by straightening. Such gaits are shown in the literature by Mori and Hirose [48], Togawa et al. [49], Ya-mada et al. [50]. Wheeled robots allow for quick movements and easy locomotion generation but the trade-off is the reduction in torsion control.

### **2.7.2 Wheel-less Snake Robots:**

Snake robots without wheels generally have 3-degree of motion. Kevin Dowling was among the first to develop a 3-D snake link composed of two orthogonal DOF. With this simple robot, he was able to obtain several gaits including sidewinding, lateral rolling, lateral undulation and non-nature-based variants of lateral rolling, ventral wave and butterfly gaits [51]. Work by Burdick et al [36], and Liljebäck et al. [42], demonstrated the ability to perform lateral undulation and sidewinding. Chirikjian and Burdick [36] [43], Ohno and Hirose [52], demonstrated caterpillar locomotion, along with lateral rolling. Many snake robots are capable of achieving similar locomotion but arrive at it through different formulations due to different kinematic models. Mathematical models are highly dependent on mechanical designs with no general model applying directly to independent mechanical implementations. Clearly, the solutions available for locomotion

pertaining to snake-like robots are diverse as demonstrated by the diverse array of solutions (even a single gait).

## **2.8 Overcoming Friction**

Snake-like robots characteristically use their body as a means for propulsion. For propulsion to occur, the body must overcome the coefficient of friction (COF). The COF is the frictional force resulting from contact made between the body segments and the surface. During the initial step of locomotion, each segment of the body is in contact with the surface, thus resulting in a high COF. To locomote, snake-like robots must generate more force than the COF created by the contact points on the body to move in the desired direction. There are three classes of approaches to address this problem: physical designs with passive casters, control algorithms and sensory based.

### **2.8.1 Physical Designs with Passive Casters**

Snake-like robots implementations often rely on physical design with the use of passive casters to locomote [15-16] [46]. Each robot segment contains an actuator and can have one or two passive casters as depicted in Figure 2.8.1A. The casters are free rolling and are strategically oriented parallel to the segment. The actuators in these snake-like robots are analogous to muscle contraction that articulates the spinal column in real snakes.

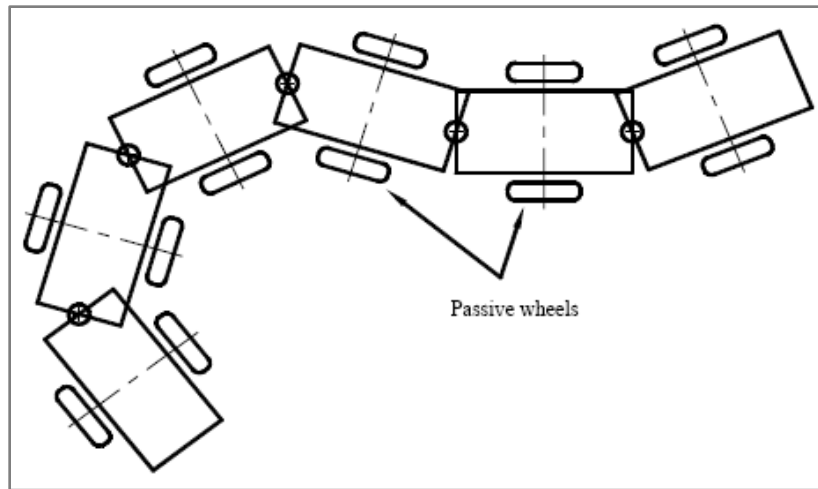


Figure 2.8.1A: Snake robot with passive casters.

Since the rotational axis for all segments are parallel, the typically achievable locomotion with this type of design is serpentine. This locomotion is characterized by wave-like undulation of the body, which creates friction that is anisotropic with respect to the direction of motion. This simply means that the transverse COF ( $u_T$ ) on each link is greater than the COF along the links ( $u_F$ ) as shown in Figure 2.8.1B.

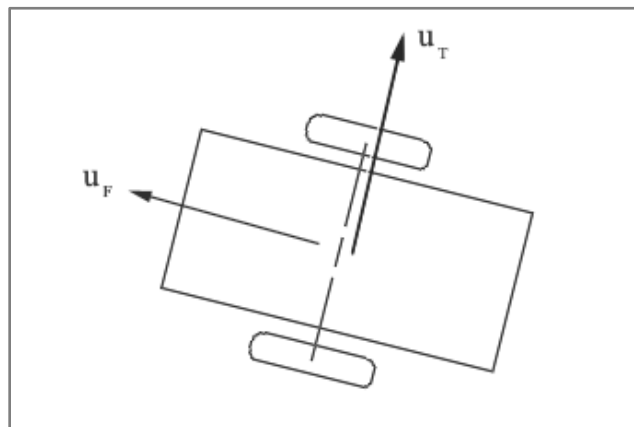


Figure 2.8.1B: The lateral friction created is much greater and forward friction:  $u_T \gg u_F$

This approach naturally has two limitations. The axis of rotation for segments are parallel linked to each other and perpendicular to the surface. This means that the surface must be planar or relatively planar. The second limitation is the coefficient of friction created by contact with the surface. Slippery surfaces such as glass, smooth hard plastic and other slippery surfaces can dramatically decrease the performance of this type of snake-like robots [34]. Surfaces with low COF can create transversal slip which decreases or inhibits forward propulsion.

### **2.8.2 Algorithmic Control Locomotion**

The second class of robots utilizes algorithmic-intelligence to address COF problem. In this case, the COF is approximated for the particular surface and the snake-like robot segments are tactically controlled to create friction in the desired section(s) of the body. The control algorithm stipulates the angle of actuation to create an increase in COF on specific section(s) while decreasing the COF in other section(s). Commonly, such robots are able to function without casters and are capable of articulating in 3-dimensions [36] [51]. A Wide range of gaits such as serpentine, lateral undulation, concertina, sidewinding and other locomotion can be achieved with this type of strategy. The downside with this approach is that the coefficient of friction is assumed to be uniform and/or the control algorithms are manually “tweaked” until they can locomote in that particular setting. These control algorithms cannot adapt to changes in the environment and result in performance deteriorations under different circumstances.

Recent developments in snake robots have shifted focus to mechanical designs. Snake robots like ACM-R5, Uncle Sam, and the Israeli military's Snake Robot have captured public attention with the release of various videos. Japan's ACM-R5 is an amphibious snake robot with fins and casters [50]. It is capable of both, land-based and swimming locomotion. In the water, it dives, surfaces and is capable of directed steering. On land it locomotes using lateral undulation and lateral rolling with the aid of passive casters. Uncle Sam is Carnegie Mellon University's latest reincarnation of their modular land-based snake robot. Uncle Sam and its variation of its linear progression gait is able to crawl on continuous terrain without the use of casters. The linear progression gait is based on their piece wise differential gait control [54] . It is also capable of climbing various poles and tree trunks based on a variation of lateral rolling described by [51].

The Israeli military's snake robot is still in its early stage of development, but already it has demonstrated the capability of crawling without wheels and is tether-less. The Israeli military has a vested interest in using snake robots in combat settings [55]. The Israeli military sees snake robot as a potential tool to be used for reconnaissance as demonstrated in one of the released videos.

### **2.8.3 Simple Sensor-based locomotion**

Earlier snake-like robotic studies primarily focused on the aforementioned class, "passive casters and algorithms", but recent research has shifted focus to locomotion based on sensory feedback from the environment. Thus the aim of such research is to encase the snake-like robot with a skin capable of a form of the sense of touch. One of



the earliest snake-like robots that possessed sensory feedback is called the GMD-snake presented in [56-58]. GMD-snake was equipped with 4 contact switches placed around the segments which are covered by a rigid shell. These contact switches allow the control algorithm to know if contact with the ground has been made. Furthermore, they also allow for knowledge of the orientation of the robot. In the event that there is longitudinal roll of the body, the state of the contact switches also changes, thus orientation can be extrapolated. Such knowledge is then used to augment actuation behaviour. The use of contact switches placed on the two sides of the segment also allows for side obstacle detection. This type of sensor has only two states: on and off, thus allowing for simpler control strategies and environmental knowledge extrapolation. The bipolar nature of this type of sensor allows the controller to only recognize contact between its segments and the ground. Nothing about the nature of the contacts such as the contact force can be determined. A different type of sensor capable of providing an input range is needed to obtain more relevant knowledge relating to the nature of the contact.

Two snake-like robots equipped with a type of sensor capable of providing contact force input ranges are Anna Konda [59] and Aiko [60]. Anna Konda is a hydraulically actuating snake robot designed for firefighting. It uses force sensing resistors (FSR) to provide a much greater sensory resolution with respect to contact switches. FSRs use the electrical property of resistance to measure the force (or pressure) applied to a sensor. Once force is applied, the FSR sensor changes its resistance this is then converted to a numerical value. This value is correlative to the amount of pressure applied onto the sensor: the value increases and decreases with respect to the force applied. Each segment

of Anna Konda is covered by 4 sensors which are protected by hard shell plates. The control strategy is to inhibit actuation once predetermined threshold values are sensed. Although FSR sensors provide a greater resolution with respect to contact switches, the use of a rigid plate covers further limit the sensing range [60]

The snake-like robot, Aiko's design was based on knowledge gained through Anna Konda. Aiko's segments are actuated via electric direct current (DC) motors instead of hydraulics and with improved FSR sensor mounts (Figure 2.8.3). Each of Aiko's segments are protected via a floating shell. This setup allows for any lateral force acting on the shell that displaces it in relation to the underlying section. Compliant material which is placed between the shell and the FSRs helps secure the shell to the segments. The compliant material absorbs some of the force which gives the effect that the shell is soft, and the sensing range is increased. The shell is also pre-tensioned to have initial bias force values. Pre-tensioning also allows for lateral force comparison between the left and right side of the segment.

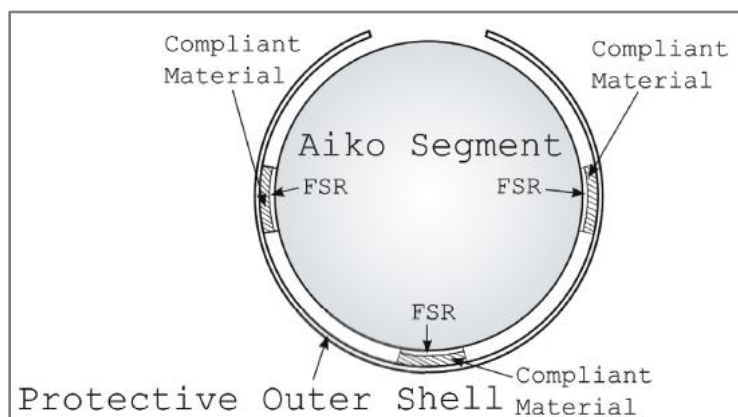


Figure 2.8.3: Aiko's segment with suspended protective shell.

The primary function of this setup is to enable lateral and ground force measurements simultaneously. Lateral force is measured as a difference of an increase of force on one side versus the decrease of force on the other side. Vertical pressure is a measurement of increasing or decreasing force input from the bottom sensor. This sensor arrangement allows Aiko to “feel” the condition of the surface and augment motor actuation base on sensory inputs. Similar to work done in [46] Aiko’s sensory setup also allows for obstacle-aided locomotion [60]. Obstacles-aided locomotion refers to locomotion as a direct result of strategically pushing against obstacles placed on the surface. The objects are strategically placed in the way of the undulation which is then felt by the sensors. In these cases the sensors are used to gain knowledge pertaining to the nature of the contacts and to measure the force generated by the push.

Locomotion based on sensory input allows snake-like robots to change their actuation behaviour with predefined threshold parameters. Sensors give snake-like robots the ability to stop from over-actuating which may result in mechanical failures. Sensors also allow other forms of locomotion that otherwise would not be possible such as obstacles-aided locomotion. This form of locomotion may provide a valuable alternative in scenarios where the surface is slippery.

## **2.9 Locomotion Control of snake-like robots**

Hirose was the first to approximate the shape snakes make while crawling through a simple model which he called the serpenoid curve [15]. Since then, work by Ma derived simple expressions for 3-D locomotion curves for a snake-like robot consisting of  $n$  rigid

links as shown by Figure 2.9. The control signal of the 3-D locomotion curve is a composition of the horizontal and vertical sinusoidal curves [61]. yaw (Z axis) and pitch (X axis) respectively as shown in figure 2.9. Equation (3) describes the joint angles for the Z and X planes of rotation for joint  $i^{\text{th}}$ :

$$\begin{cases} \theta_i(s) = -2\alpha_{\theta 0} \sin\left(\frac{K_n\pi}{n_\theta}\right) \cdot \sin\left(\frac{2K_n\pi}{L}s + \frac{2K_n\pi}{n_\theta}i\right) \\ \phi_i(s) = -2\alpha_{\phi 0} \sin\left(\frac{K_n\pi}{n_\phi}\right) \cdot \sin\left(\frac{2K_n\pi}{L}s + \frac{2K_n\pi}{n_\phi}i + \delta\phi\right) \end{cases} \quad (3)$$

Equation (3): Both,  $\theta_i$  is the yaw angle and  $\phi_i$  is the pitch angle for the  $i^{\text{th}}$  link. The equation describe gait motion in 3-D space.

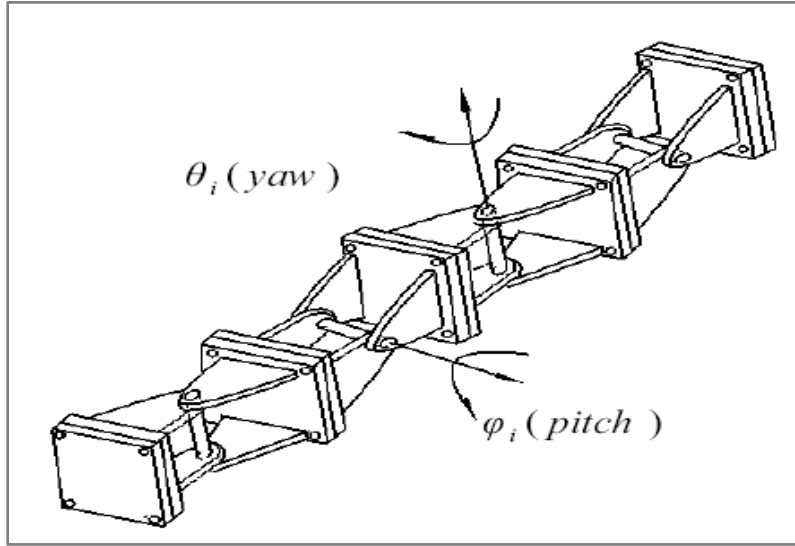


Figure 2.9: Model of snake robot joints configuration

The rotation around the pitch axis is represented by  $\phi_i$  and  $\theta_i$  describe the rotation around the yaw axis. Variable  $\alpha_{\phi 0}$  and  $\alpha_{\theta 0}$  stipulated the initial winding angles on each plane, and  $n_\phi$  and  $n_\theta$  are the number of links pertaining to each plane. Variable S is the

displacement of the tail along the serpenoid curve path, which determines the changing frequency of the body curve.  $K_n$  is the number of wave shape,  $L$  is the total body length of the robot and  $i$  is the respective link on the robot. The variable  $\delta\theta$  is the phase difference between two waves of phase.

### 2.9.1 Serpentine gait

Serpentine gait is a typical form of locomotion among most snakes. This form of locomotion is able to traverse slippery surfaces, hence it is widely applicable to most of surfaces found in nature. Work by [61][62] describe the serpentine movement in terms of varying  $\theta_i$  (yaw) angles and hold the  $\varphi_i$  (pitch) constant. The serpentine movement is given by:

$$\theta_i(s) = -2\alpha_0 \sin\left(\frac{K_n\pi}{n}\right) \cdot \sin\left(\frac{2K_n\pi}{L}s + \frac{2K_n\pi}{n}i\right) + K_1l \quad (4)$$

Equation (4): Input variables are :  $\alpha$  , s,  $K_n$  , n, l

Where  $K_l$  is the additional bias curvature for turn and  $l$  is the length of each link. For a 3-D snake robot the link is composed of both yaw and pitch rotation which may result in two sub-links, otherwise it is the length of one link. With this formula the amplitude, phase and frequency can be directly changed to generate locomotion.

### 2.9.2 Concertina gait:

Concertina movement is an effective form of locomotion in narrow passages and climbing. This locomotion is produce forward movement via a series of alternating

pulling and pushing of the body segment into relatively small set of folds (Figure 2.5.2A -Figure 2.5.2B). The bends may laterally push against the side of tunnels or vertically against the ground thus keeping the body from slipping. The motion is created when the rotation angle in the yaw ( $\theta_i$ ) axis is a constant and the pitch ( $\phi_i$ ) angles are varying. Concertina movement [63] is obtained by:

$$\phi_i(s) = -2\alpha_0 \sin\left(\frac{K_n\pi}{n}\right) \cdot \sin\left(\frac{2K_n\pi}{L}s + \frac{2K_n\pi}{n}i\right) + K_1l \quad (5)$$

Equation (5): Input variables are  $\alpha$ ,  $s$ ,  $K_n$ ,  $K_1$ ,  $n$ ,  $l$ .

### 2.9.3 Sidewinding gait

Sidewinding gait is typically employed by snakes in the desert or on surfaces with low shear. It was first described by [36]. During locomotion only 2 to 3 parts of the snake touch the surface at any time. The gait is driven by 3-D rolling of each joint without contact friction compensation found with lateral undulation. The body curve of this gait is described as the composition of rotation for both X and Z axis with a phase difference,  $\delta\theta$ . Sidewinding gait [62] is generated by equation (6):

$$\begin{cases} \theta_i(s) = -2\alpha_{\theta_0} \sin\left(\frac{K_n\pi}{n_\theta}\right) \cdot \sin\left(\frac{2K_n\pi}{L}s + \frac{2K_n\pi}{n_\theta}i\right) \\ \phi_i(s) = -2\alpha_{\phi_0} \sin\left(\frac{K_n\pi}{n_\phi}\right) \cdot \sin\left(\frac{2K_n\pi}{L}s + \frac{2K_n\pi}{n_\phi}i + \delta\phi\right) \end{cases} \quad (6)$$

Inputs Prameters: :  $\alpha_\theta$ ,  $\alpha_\phi$ ,  $\delta\phi$ ,  $s$ ,  $K_n$ ,  $n_\theta$ ,  $n_\phi$ ,  $L$ ,  $i$

#### 2.9.4 Lateral Rolling

Lateral rolling is a gait not found in biological snakes but it is a simple and effective gait for snake-like robots first demonstrated by Kevin Dowling [51]. Lateral rolling gait requires the robot to periodically flex into an arc and straighten resulting in lateral movement. This locomotion is also a 3-D movement but without the undulation found with sidewinding. The body curve of this gait is described as the composition of rotation for both X and Z axis and the phase difference ( $\delta\theta$ ) between every two joints is zero [42]. The lateral rolling gait [64] is described by Equation (7):

$$\begin{cases} \theta_i = -2\alpha_{\theta 0} \sin\left(\frac{K_n\pi}{n_\phi}\right) \cdot \sin\left(\frac{K_n\pi}{L_\theta} s\right) \\ \phi_i(s) = -2\alpha_{\phi 0} \sin\left(\frac{K_n\pi}{n_\phi}\right) \cdot \sin\left(\frac{2K_n\pi}{L_\theta L} s + \delta\phi\right) \end{cases} \quad (7)$$

Input Parameters:  $\alpha_\theta$ ,  $\alpha_\phi$ ,  $\delta$ ,  $s$ ,  $K_n$ ,  $n_\theta$ ,  $n_\phi$ ,  $L$

#### 2.10 Summary and analysis

This chapter has presented numerous ways of generating Snake-like locomotion, but the afore-mentioned method described by Ma et al, allows for simple computation of joint variables that does not require in-depth numerical analysis, unlike ones based purely on kinematic modeling. These expressions can easily be computed by most modern microcontrollers. Several snake robots were discussed with different design approaches as well as control methods. Snake robots with all their advances over the past 30 years have not been used in real-world settings. Their main attribute of adaptivity due to multi

actuating segments is also their greatest hurdle to progress. The complexity of controlling multi-segments in ways that give snake robots the capability to automatically overcome disjointed surfaces has not been effectively addressed as yet.





## Chapter 3

# Striker and Gait Training Method

### 3.1 Introduction

Land-based mobile snake robots are complex systems that cannot merely be evaluated through single tasks or simulations alone. Real systems must be understood in the context of the real world and in comparisons to similarly purposed systems. In this chapter we present our very own snake robot called, “Striker”, its hardware design and architecture. Secondly this chapter presents a novel solution for creating new gaits, on-the-fly with the Explicit Gait Training (EGT) method which we apply to Striker. Thirdly the Standard Mobility for Snake Robot (SMMSR) is presented as a means to evaluate the effectiveness of snake robot locomotion.

### 3.2 Striker’s Metal Skin

Striker was modular and tethered snake robot developed at the N-CART laboratory based on the knowledge gained from Prototype S1 as shown in Figure 3.2A. The Prototype S1 was a quick conceptual model we developed to re-evaluate findings from the literature. Prototype S1 was assembled using plastic joints and without a body casing. From prototype S1 we concluded that a protective housing, *skin*, was essential to cover the actuators and protect the wire connections associated with any practical snake robot.



Figure 3.2A: Prototype S1 with links connected with plastic brackets.

The skin also allows for more uniform surface contact along each individual segment. Secondly, the joint between segments needed to be more robust and allow for wider pivoting angle ranges. A wide pivoting angle is a defining feature for snake robots that allows for greater range of motions for each segment. Greater range of motion by individual links allows the entire snake robot to be more adaptable and to locomote faster [34].

After several designs and different construction material we selected a light 6061 aluminum alloy. The aluminum sheets are relatively light and malleable enough to be folded but still retain their structure with repetitious impacts during locomotion. Each housing and joint bracket is designed using a single folded piece of aluminum sheet as shown in Figure 3.2B. Together, the housing and joint brackets constitute a segment, or two links.

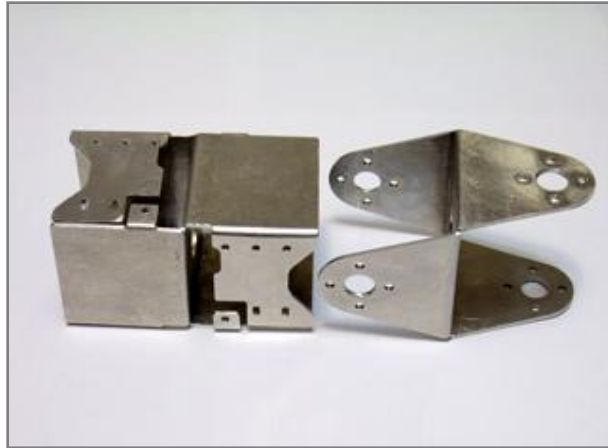


Figure 3.2B: Striker's metal skin: actuator housing and joint bracket. See Appendix A for the unfolded CAD design of the brackets.

### 3.3 Striker the snake robot

Striker is a tethered snake robot composed of seven segments and one NSR microcontroller as shown in Figure 3.3. Unlike snake robots such as [15], [46], [50], Striker is wheel-less. Each segment contains two smart module actuators to allow for a 3-D range of motion. The smart actuators are positioned with a  $90^{\circ}$  rotations with respect to each other; one actuator for the pitch axis and another actuator for the yaw axis. This configuration is similar to that found in other snake robots such as [51], [59], [60], [64]. Each segment is 13.9cm long giving Striker a total body length of 97.3cm. The design of each segment allows  $190^{\circ}$  of rotary motion along each joint;  $90^{\circ}$  rotation to the left and  $90^{\circ}$  rotation to the right with respect to the top of the smart actuator. Due to high power demand from the smart actuators, Striker is powered via tether power source.



Figure 3.3: Striker; composed of one NSR microcontroller and 7 modular segments.

### 3.4 Striker's Communication Architecture

The NSR microcontroller and the smart actuators communication architecture is akin to that of a simple multi-drop bus network [65]. The NSR controller communicates with smart actuators by asynchronously sending and receiving data packets. Asynchronous data transmission reduces the chance of data collision, and can attain transmission rates up to one megabaud. With the exception of the first and the last smart actuator, all actuators are linked to two others. This contiguous chain topology means data packets sent from the NSR controller are read by all actuators as shown in Figure 3.4.1.

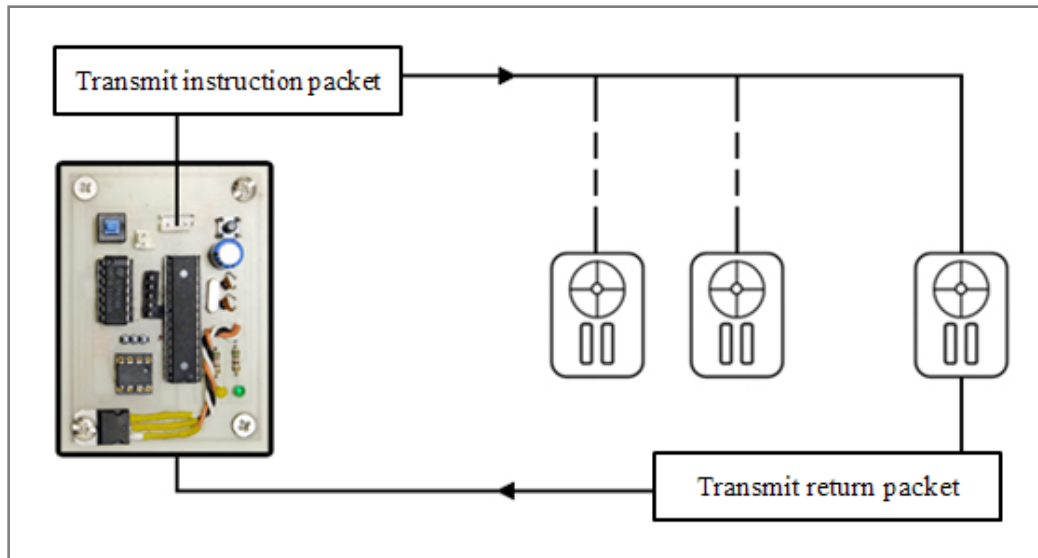


Figure 3.4.1: Multi-drop interconnection and data transmission. Each smart actuator is like a little node on a bus network.

The data packets are simplifications of a network communication protocol [65]. The data package begins with a two-bytes header, one identifier byte, instruction byte, parameter byte(s) and end with a checksum (Figure3.4.2).

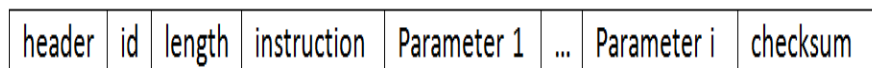


Figure 3.4.2: Packet structure: a collection of unsigned bytes of value from 0 to 255.

Each packet column value ranges from 0 to 255. The header values are constants with values of 255, the identifier, id, byte value signifies which smart actuator will execute instruction with the embed parameter(s)<sup>3</sup>.

The NSR microcontroller initiates communication by transmitting a data packet with a particular id, instruction and the pertaining parameter(s). The packet is read by all smart actuators but only the unit with the matched identifier will transmit an acknowledgement or a respective packet back. The packets sent by the NSR microcontroller can be classified into two classes that are similar to the Set and Get functions in programming. The set instructions commanding the actuator(s) to carry out a physical task such as rotate to a particular angle or to change operation parameters such as torque limits or actuating speeds. The Get instructions are ones telling the actuator to report about its current state such as torque value and temperature. The return packet has the same structure as the instruction packet. If the packet was in the form of a Set command from the NSR, then the actuator will reply with either an acknowledge or error packet. If the packet is in the form of a Get command then the respective actuator will send a packet back with the requested information.

### **3.5 NSR Micro-controller**

The NSR microcontroller hardware and software architecture is relatively simple due to the modularity of the smart actuator units. The NSR unit is where all joint angles are computed for the gait and logics for inputs data reside. Most secondary tasks are

---

<sup>3</sup> Consult Dynamixel-AX12's manual at [www.robotis.com](http://www.robotis.com) for further details.

allocated to the smart actuators. Secondary tasks such as motor control, threshold variables such as torque limits, over heating settings are innate to the smart actuator. A non-centralized architecture such as this one allows the software and physical design of the main controller to remain small and thus efficient. The main controller is primarily responsible for computing the shapes needed to generate locomotion, and transmit instructions to smart actuator units for execution into movement.

The NSR microcontroller was built out of necessity. We are not making a claim about the research contributions of this development but it was a necessary step in this project's evolution. At the inception stage of the project there was no alternative micro-controller to the CM-5 sold by the manufacturer. Secondly, the CM-5 came with a visual programming interface that was not sufficient to handle the task intended and no software library was available to allow direct communication with the smart actuator, Dynamixel AX-12. We reverse-engineered the smart actuator communication protocol and created our own library using the C programming language<sup>4</sup> and finally designed a micro-controller called "NSR" (Figure 3.5A).

---

<sup>4</sup> See Appendix B for the code library



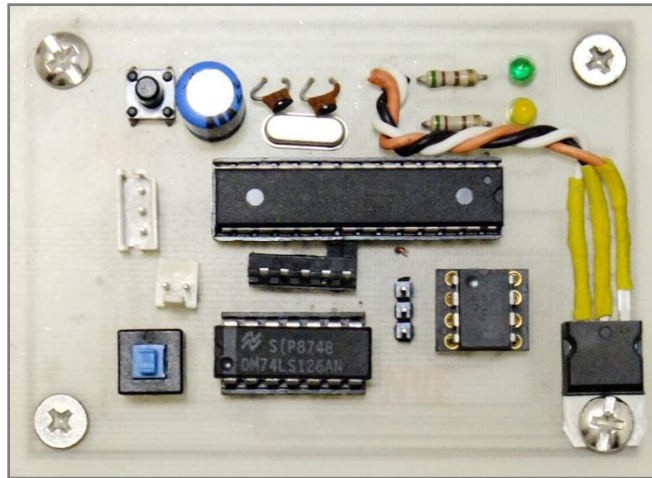


Figure 3.5A: NSR microcontroller with tri-state buffer. See appendix C for complete schematic layout of the NSR and communication packet Library.

The NSR hardware and software architecture is relatively simple due to the modularity of the design. The NSR is analogous to the head of Striker; it is where the gaits' joint angles are calculated. The NSR microcontroller was designed using an Atmel Atmega8 with a tri-stage buffer tied to the transmit and receive lines. The Atmega8 was later replaced with the Atmega168 to double the processing speed and memory storage. The tri-stage buffer is needed to allow asynchronous communication with the smart actuators. There is no motor driver on the NSR controller because each smart actuator has a built-in microcontroller and motor driver. The complete specification of the NSR microcontroller can be found in Table 3.5.

Table 3.5: Specification of NSR microcontroller for Striker.

NSR Specification	
CPU	<i>Atmel Atmega168</i> <i>16KB Flash Memory</i> <i>512 EEPROM</i>
Clock Crystal	<i>16MHz</i>
PC Serial communication	<i>Max245</i> <i>Data rate 120kbps</i>
Tri-state buffer	<i>Fairchild DM74LS126AN</i>
Indicators	<i>2 LEDs: Power and Status</i>
Switches	<i>Reset &amp; power</i>
Connection Header	<i>Molex 22-03-5035:</i> <i>3 Pins header use to communicate and power smart actuator</i>
Power header	<i>2 Pins power header</i>
Regulator	<i>5V to microchip circuit</i>
Input Voltage ( V )	<i>9.6V-12V</i>
Output Voltage ( V )	<i>9.6V-12V:</i> <i>Unregulated output to Molex header</i>
Dimension	<i>8.0cm x 6.0cm</i>

The NSR's primary roles are to discretize gaits, calculate joints angles, transmit instruction packets to each smart actuator and read the feedback packets. The NSR microcontroller generate the time variant shapes of the snake-like robot that will allow Striker to locomote in the manner that satisfies specified conditions. The shape is discretized into 14 equal links. The discretization process creates angles for all links. The output angels are in radians and must be converted to value usable by the smart actuators. The smart actuators produce position values from 0 to 1024 covering 300 degrees which means one degree is equivalent to 3.4133333 actuator value. The actuator is centred at the 511th position value or 150 degrees. The following procedure shows how radian angles are converted to actuator position values (Procedure 3.5):

1. Radian to Degree Conversion:

$$\varnothing^0 = \varnothing^c \times \frac{180}{\pi}$$

Where  $\varnothing^0$  &  $\varnothing^c$  denote degrees and radians, respectively.

2. Due to the orientation of the smart actuator in Striker the angles outputted are perpendicular to the middle of head of the smart actuator, 150 degrees with respect to the smart actuator position, the following need to transform the angle to the proper value for the smart actuators:

If  $\varnothing^0 \leq 60$

$$AX\_angle = 60 - \varnothing^0;$$

If  $\varnothing^0 \geq 120$

$$AX\_angle = 240 + 180 - \varnothing^0$$

3. Convert AX\_angle to goal position value:

$$\text{Position value} = Ax\_angle \times 3.4133333$$

Procedure 3.5: Convert radian angle to smart actuator position value.

Data packets are sequentially composed for each id with its respective angle. Using the communication protocol, the NSR controller transmits the packet and waits for the smart actuator identified in the packet to send a packet back. In the event that the return packet

is returned and contains no error message, then the main controller proceeds to transmit the next packet. In the situation where the return packet is not received in a timely manner or the return packet contains an error message, then the main controller resends the same packet. The algorithm that governs communication in the NSR controller is relatively simple as shown in Figure3.5B:

```

NSR Microcontroller
While(true)
{
  Generate body shapes (t)
  Discretization
  Form packets
  for(j<= number of links)
  {
    Transmit packet(i)
    Read returnpacket(i)
    If no packet return{
      Transmit packet(i)
    }
    Else{
      Decode returnpacket
      if(error==returnpacket message)
        Transmit packet(i)
    }
  }
  Delay(ms)
}
.....
}

```

Figure 3.5B: NSR Algorithm to generate gate based on mathematical formulation.

### 3.6 Smart actuator

Snake robots, with their multitude of actuators, create situations where cable management inevitably becomes a logistical nightmare, especially when the wires are long. Actuators

are typically motors and servos. Whether the motors are brushed, brushless or stepper types, they need a special motor driver circuit to regulate their current flow. The microcontroller controls the motor driver. In the case of servos they need a microcontroller capable of generating pulse width modulation signals. These types of actuators require direct connections back to the microcontroller or motor driver(s). This architecture would have resulted in 42 cables being connected to the microcontroller for Striker. Therefore these kinds of actuators are not suitable for contiguous configurations such as exhibited by robotic snakes.

The smart module actuators are Robotis' Dynamixel AX-12. Each smart actuator has its own built-in Atmel Atmega8 microcontroller, motor driver, rotary motor with a position and thermal sensor. These actuators are capable of position, torque load and temperature feedback. The position output range from 0 to 1024 values covering 300 degrees, and it is centred at value 511 as shown in Figure 3.6.1.

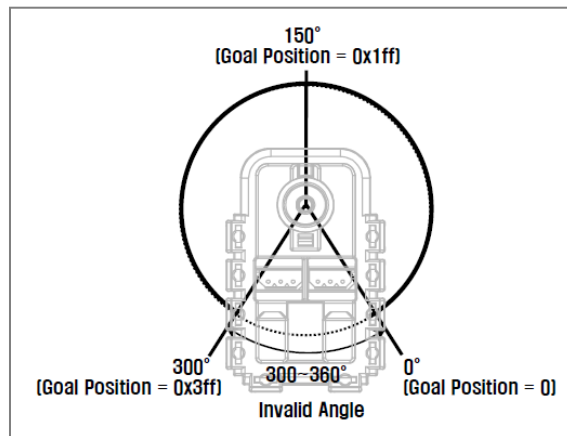


Figure 3.6.1: Smart actuator position value and possible valid angle.

There are two identical 3-pin Molex ports situated on the unit that allow for ground, power and data communications. With the exception of the first and last smart actuators, these ports allow two additional smart actuators to be connected. This means only one smart actuator on the chain needs to be connected with the NSR microcontroller (Figure 3.6.2). The multi-drop connection means each smart actuator must have a unique identifier in order to identify itself to the main controller. When a packet is received by the actuators, only the one matching the id will send an acknowledgement packet and perform the action.

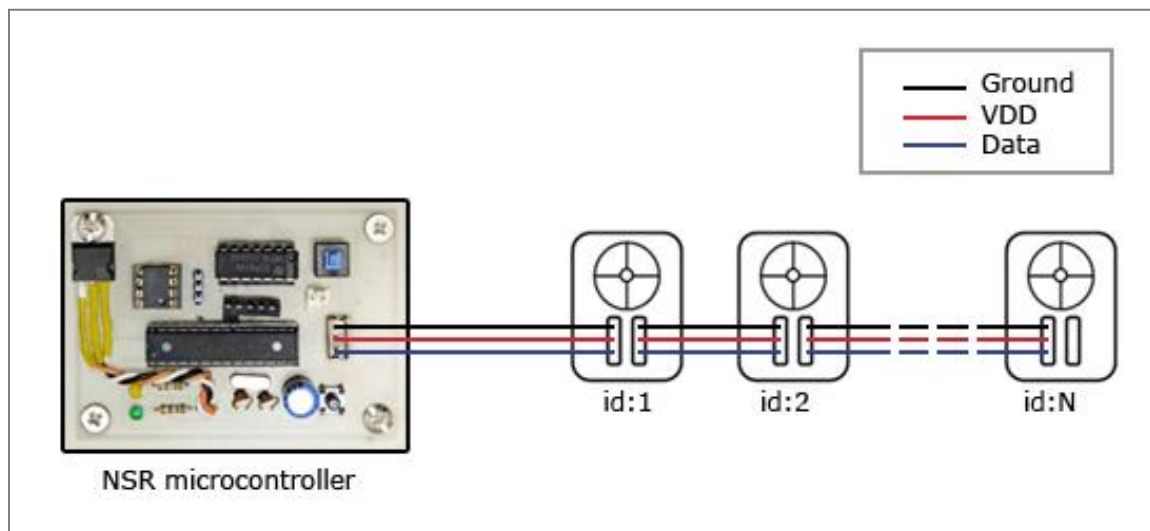


Figure3.6.2: The NSR Microcontroller needs only three wires to communicate with all smart actuator.

The smart actuators also have built-in self-preservation features, such as auto shutdown, when its parameters are outside of the norm. Each unit is in a state of constant monitoring of parameters such as temperature, torque overload, invalid speed and

position. These parameters are governed by predefined threshold values. Smart actuators perform only within the predefined threshold. Once a parameter is outside of the predefined range the smart actuators will terminate their actions, and send a packet containing the respective error message to the NSR microcontroller. The Algorithm governing the smart actuator is depicted by Figure 3.6.3.

```

Smart Actuator
While(true)
{
  Read packet
  Check if packet is valid (checksum)
  If (self id == id){
    Sent acknowledge packet
    Perform instruction
  }
  else
    no action
}
....
}

```

Figure3.6.3: Smart actuator algorithm.

### 3.7 Explicit Gait Training Methodology

Snakes are very versatile animals capable of living in almost all terrains found on Earth, and snake robots certainly have the potential to achieve similar capabilities. Snake robots, unlike real snakes are limited by the gaits that are preprogrammed in them. Snake robot design may be simple but gait controls are often very complicated since they are made up of multiple segments connected by joints which can possess up to 5-DOF. Striker consists of segments that have 3-DOF. Mathematical control algorithms must



precisely generate specific joint angles for all the joints at each time step in order to create directed locomotion. Control algorithms such as ones in Chapter 2 have many input parameters that must be specifically tuned based on each individual snake robot and for a particular type of surface. From first-hand experience the tuning process is time consuming and can take up days, if not longer, to get the desired outcomes for a particular surface. It is immediately apparent that one cannot tune control algorithms to account for all types of terrains. Terrains with disjointed surfaces such as ones with different elevations and/or gaps present the greatest challenge to snake robots. In situations such as typically found in urban disaster environments where time is often a critical factor and the terrain topography is inevitably unpredictable, one does not have the luxury of delay in order to tune the control algorithm when the snake robot is stuck or at an impasse. Currently there exists no snake robot with the necessary intelligence, i.e., control algorithm(s) and/or mechanical design capable of resolving itself from a stuck situation or conquering disjointed surfaces automatically. Until these limitations are overcome, snake robots will remain merely academic challenges with no real application.

The Explicit Gait Training Method is an elegant solution that allows for the creation of new snake gaits very quickly. The EGT method is most suited to resolving any impasses that a snake robot may encounter during operations. The EGT method allows for the design of new gaits specifically tailored to the situation at hand. The EGT method is

based on three simple, but vital points:

1. Repeatable motion,
2. Human Expert Knowledge and
3. A simplified gait creation process.

The EGT method allows the creation of a bevy of novel gaits outside of laboratories and research settings. In addition, this method supports, perhaps for the first time, trial-and-error testing of snake gaits.

The first point takes advantage of the characteristic of snake gaits. Snake gaits are composed of repetitious motions while keeping their overall distinct shapes. These traits are observed with serpentine, sidewinding and concertina gaits presented in section 2.5. We can use this principle to create new gaits. Given that the actuation speed of each joint is kept constant, we only need to capture key characteristics between movements. This is akin to animation where only key frames are manually created and the motion is the generated frame transitioning from one key frame to the next, also known as “tweening” [66]. Using this principle, the complexity of any gaits can be decomposed into only a few key frames.

The second point emphasises the human advantage. Human beings, when faced with a situation, can process and analyze information more quickly and accurately than any other artificial systems. Hence, we use the experience of robots operators or first responders and their knowledge of the terrain to create new gaits. They possess a vast

amount of information about terrain as they can observe it on scene. With practice and training on how the robot operate in different situations, first responders and robot operators can combine their knowledge of robots' operational behaviours and knowledge of the terrain to apply them to specific situations. With the information pertaining to the situation already processed, robot operators apply their knowledge and experience to create the key frames or poses that exploit characteristics within the terrain. The result of the EGT method is a unique gait, quickly generated, to address a novel situation.

Thirdly, snake-like robotics has been a subject of study for over 30 years but they have not yet moved beyond the confines of the academic arena. Snake robots are fickle systems due to their high dependency on physical designs and control algorithms that must be tailored to each robot and the interacting surface. Traditional laboratory-based methods of creating and designing new control laws such as presented in chapter two require a very large amount of time to achieve desirable results. The EGT method allows nearly anybody to transfer their knowledge and expertise to create new gaits quickly. The EGT method gives operators a means to compensate for any locomotive limitations of traditional gaits as well as utilize the full extent of the flexibility and potential of snake motion. Once a gait is created using this method, it can be archived to create a library of different gaits.

### **3.8 EGT Algorithm**

The EGT algorithm begins with the trainer's assessing the current terrain impasse. Once the assessments are complete, the training process for a new gait begins. To train a new

gait the trainer must create a set of successive poses of the snake robot at each time step as shown in Figure 3.8.

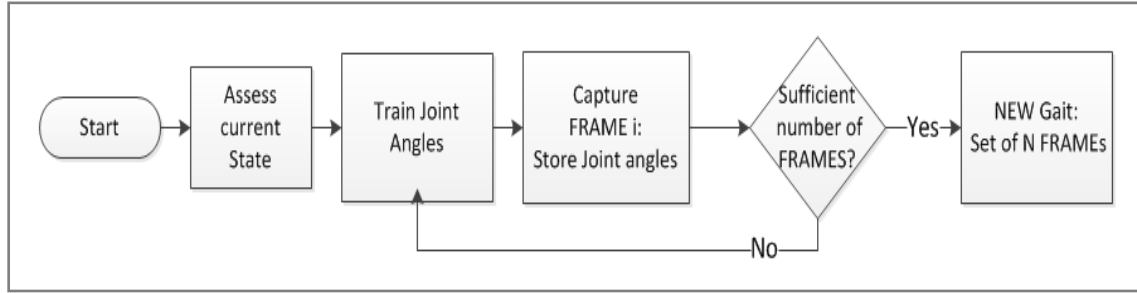


Figure 3.8: Explicit Gait Training Algorithm

Each pose requires every joint angle value to be explicitly trained and captured. This is done by physical manipulation, using the snake robot as a programming interface. The set of joint angles ( $q_i$ ) captured for each pose is referred to as a “FRAME”. An example for a FRAME for Striker is the following:

$$FRAME_i = \{ q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}, q_{11}, q_{12}, q_{13}, q_{14} \}.$$

In the case where the FRAME contains only actuator position values of 511, Striker is perfectly straight. The trainer continues to train successive FRAMES to incorporate all desired attributes, such as shapes and orientations. Once a set of N number of FRAMES is established, it can be executed in sequence as a new gait with distinctly trained characteristics.

During the training process the trainer does not need to be concerned about the position value of each actuator. The trainer’s primary goal is to affectively apply his/her

knowledge of the terrain to train the joints to capture the desired configuration in each FRAME. In cases where a FRAME is not suitable, it may be discarded and quickly replaced by a new FRAME. The same can be said about the newly trained gait. Once a gait is trained, it can be stored for future use. Consequently, one could have a library with an assortment of trained gaits.

### **3.9 Standard Mobility Metric for Snake Robots**

The SMMSR arose from the need for transparency when evaluating and reporting snake robots' locomotion abilities. The SMMSR allows assessment of snake robots' locomotion in controlled settings but with surfaces that are analogous to aspects of terrains found during urban search and rescue operations within disasters formed through the structural collapse of buildings.

One of the goals for snake robots is to be used for search and rescue situations such as collapsed buildings and structures [67-70]. Observations during the 2008 National Institute of Standards and Technology (NIST) Response Robot Evaluation Exercise, and 2006 to 2008 Heavy Urban Search and Rescue (HUSAR) exercises and many other Ontario Provincial Police (OPP) training exercises found that topographies in collapsed buildings and structures are random and chaotic. Topographies of search sites are unique in that they depend on the building materials and structural layout of the buildings. There is no known method to model topography for disaster-based collapsed structures and buildings.

Unlike other land-based robots, most of a snake robot's body needs to have constant contact with the surface to generate directed motion, and changes to surface topography greatly affect the way snake robots can locomote. Thus a standard method of evaluating how well snake robots move on a surface is important for understanding and advancement of snake robots in general and their mobility specifically.

The SMMSR is a proposed base metric that consists of five tests made up of different surfaces: Planar Mobility Test (PMT), Incline Mobility Test (IMT), Narrow Passage Mobility Test (NPMT), Obstacle Mobility Test (OMT) and Step Mobility Test (SMT). The SMMSR does not exhaustively cover all surfaces but rather the tests are decompositions of surfaces commonly found in collapsed structures and buildings. The tests take their inspiration from proposed test methods used for testing the mobility of response robots [27] [33]. The SMMSR is a tool which may serve as a standard platform to qualitatively evaluate and quantitatively benchmark different land-based snake robots; it provides a means to bring transparency to report findings and comparisons of different snake robots' abilities. We use SMMSR to evaluate Striker's ability to traverse different surfaces employing different gaits; we measure speed and report qualitative observations.

SMMSR surfaces are constructed using common materials, readily available at hardware stores. Our SMMSR surfaces are built using various common materials.

### **3.9.1 Planar Mobility Metric**

The Planar Mobility Test (PMT) is the first test in the SMMSR set. It is the simplest and most fundamental test surface and is composed of a planar flat surface typically found on

most laboratory floors and/or table tops as shown in Figure 3.9.1. The PMT serves as a base test where standard snake gaits such as serpentine, concertina, sidewinding and lateral rolling can be quantified.

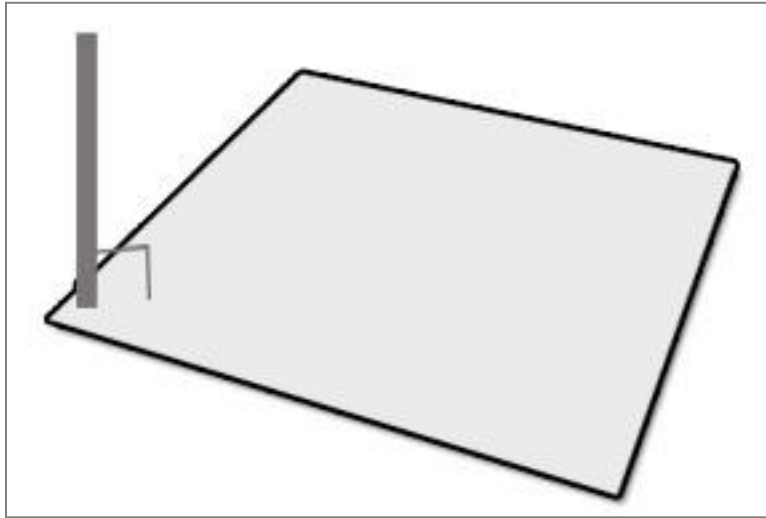


Figure 3.9.1 Planar Mobility Test the base test for all land base mobile robots.

### 3.9.2 Incline Mobility Test

The Incline Mobility Test (IMT) consists of a flat hard surface situated on an angle of  $\Theta$  degrees as shown in Figure 3.9.2. Inclined hard surfaces have a lower coefficient of friction, which drastically reduces or completely impairs the mobility of robots. Inclined surfaces are common in most environments including those in search and rescue operations. With the IMT we can measure the inclined angle and locomotion speed that is achieved.

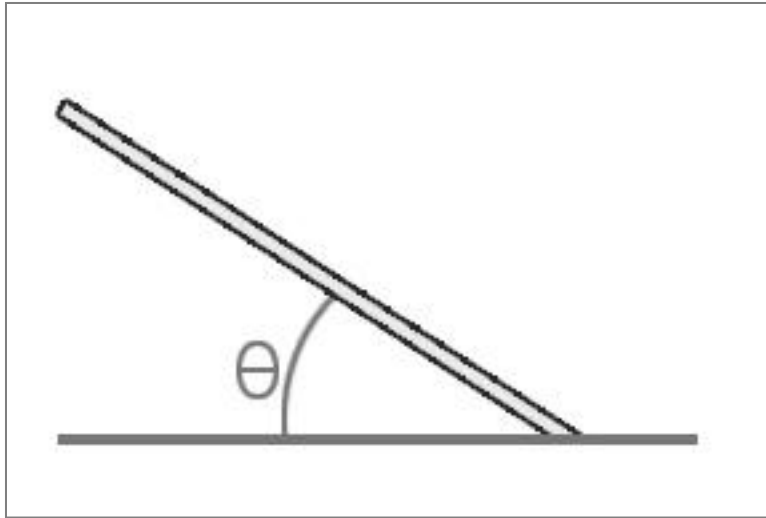


Figure 3.9.2 Incline Mobility Test;  $\Theta$  degree inclined angle

### 3.9.3 Narrow Passage Mobility Test

The Narrow Passage Mobility Test (NPMT) evaluates a snake robot's ability to traverse through environments with confined spaces, such as cracks or narrow passages. With the inherent slender design of snake robots, NPMT tests this inherent advantage within the class of snake robots allowing a comparison akin to “apples to apples”. The NPMT tests the potential of snake robots to traverse small cavities and narrow passages that are prohibitive for other mobile robots. The NPMT can be constructed with a closed or open top passage as shown in Figure 3.9.3. We constructed the NPMT using wood with an open top and an adjustable sidewall. We investigated Striker's ability to traverse the narrowest possible passage and measured its speed.



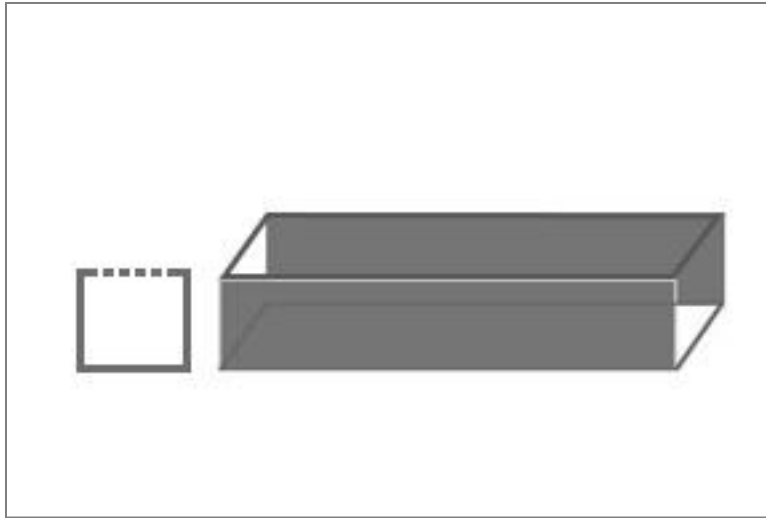


Figure 3.9.3 Narrow Passage Mobility Test; Closed or open top

### 3.9.4 Obstacle Mobility Test

The Obstacle Mobility Test (OMT) tests a snake robot's ability to overcome obstacles. Snake locomotion pushes against a surface to move, which means its segments are situated on a surface or very close to it; thus, obstacles present an abrupt change to the surface that may impede mobility. One constraint in this test is that the obstacle used for testing can be anything but must be easily replicated. As an example of the OMT, a rectangular block serves as an obstruction that needs to be overcome as shown in Figure 3.9.4.

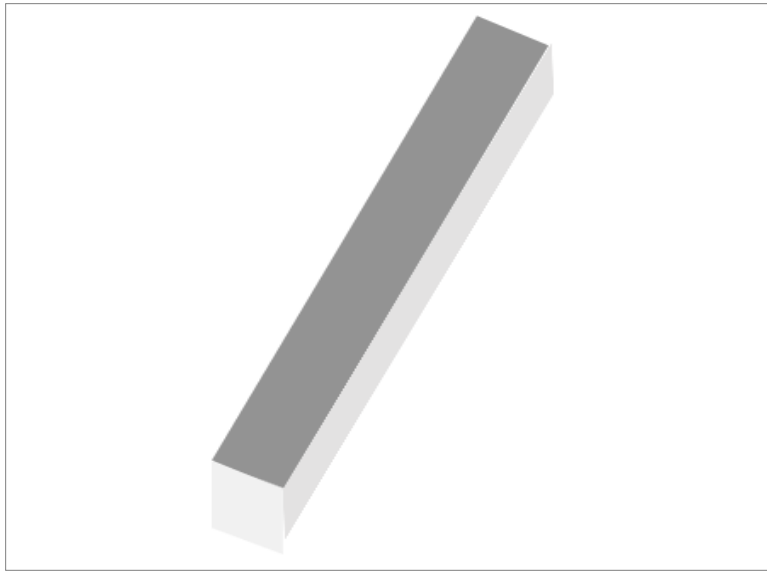


Figure 3.9.4 Obstacle Mobility Test: Can use any shape obstacle

### **3.9.5 Steps Mobility Test**

The Steps Mobility Test (SMT) tests a snake robot's ability to climb stairs as shown in Figure 3.9.5. Stairs are a common feature of almost all human-built environments. Their presence can safely be assumed for all tasks envisaged for a snake robot. During the 2009 NIST Robot Exercise and our own evaluation with the OPP robot, we observed that climbing stairs is a daunting task for tracked and wheel-based mobile robots. We constructed SMT using wood to evaluate Striker's ability to climb stairs.

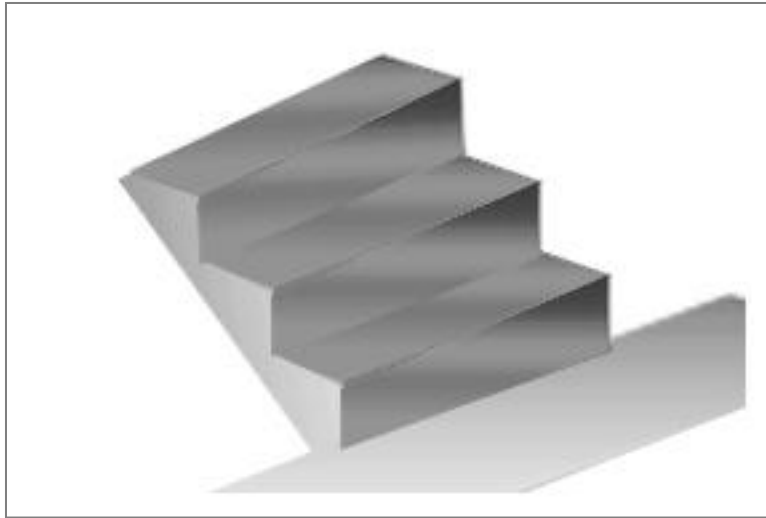


Figure 3.9.5 Steps Mobility Test

### 3.10 Summary

The snake robot, Striker, was modularly designed to allow it to be robust in the face of mechanical failure. The housing and joint bracket is folded from a single sheet of aluminum alloy providing both simplicity and durability. The effort was not lost because the resulting parts are robust and therefore reduce the likelihood for the need for additional parts. The modular architecture of Striker allows the SNR microcontroller design to remain relatively simple, including its software architecture.

The EGT method presented in this chapter allows almost anybody to train new gaits for snake robots to address the situation at hand. This method allows robot operator(s) to apply their knowledge of the terrain to create new gait(s) for snake robots. The EGT

method allows snake robots to be adaptable when the preprogramed gaits are not sufficient to traverse the terrain.

The Standard Mobility Metric for Snake Robots (SMMSR) is proposed as a base test beds use to evaluate the effectiveness of snake robots. This allow for transparency when comparing different snake robots' capabilities. Striker will be the first to be evaluated using the SMMSR.



## Chapter 4

# Experiments and Results

### 4.1 Introduction

Evaluating an actual system via a tangible test bed, as opposed to investigating through simulations, is a proven method to provide real data that allows for a more reliable understanding of system behaviours in predefined, repeatable settings. Simulation is a good tool to prime initial conditions when the system dynamics are modeled with a great deal of accuracy. However, accurate models are often very difficult to achieve due to the complexity of the actual conditions and the limitations of simulation tools. Testing actual snake robots with a real test bed is especially important because of the complexity in the robot itself and the randomness of the intended terrains such as ones found within urban disasters.

This chapter presents experimental results of various gaits from mathematical control models and gaits created through the EGT methodology on SMMSR's five different test surfaces. Experimental results were obtained using the snake robot Striker.

Just to “throw a wrench in the works” so to speak, the Director of research at N-CART, Dr. Alex Ferworn, issued a challenge to test Striker and the EGT method in the field on

the exact terrain where an established CBRNe<sup>5</sup> variable geometry robot had failed before. We will discuss the challenge, the process and the outcome in the latter part of this chapter followed by an analysis of Striker's performance.

## **4.2 Experimental Results**

### **4.2.1 Experimental results for PMT**

Serpentine gaits described by equation (4) in chapter 2 generated no desirable forward motion on PMT. The formula for this gait created resultant motion that is parallel to the surface. The control law works with snake robots which are fitted with passive casters that have a rotational axis parallel to their segments. This allows for the coefficient of friction perpendicular to the links to be greater than the coefficient of friction along the links, which in turn propels the robot forward. Without casters Striker's coefficient of friction is the same in all directions; thus, it was observed to wiggle in one place with no productive forward motion.

With Concertina Gait using Equation (5) with the following input parameters  $\alpha = 0.5$ ,  $s = 10$ ,  $K_n = 2$ ,  $K_l = 0$ ,  $n = 7$ ,  $l = 14$ , Striker was observed traveling forward at a rate of 1.6 metres/minute. Other variations of  $\alpha$ ,  $s$ ,  $K_l$  were tested but found to be unstable and/or did not perform very well.

With the Sidewinding gait we observed Striker moving at 1.65 metres/minute at approximately  $50^\circ$  to  $60^\circ$  with respect to a forward facing direction using equation (6). The input parameters used were  $\alpha = 0.4$ ,  $\delta = \pi/4$ ,  $s = 10$ ,  $K_n = 2$ ,  $n_\theta = 7$ ,  $n_\phi = 7$ ,  $l = 14$ . Two

---

<sup>5</sup> Chemical, biological, radiological, and nuclear explosive (CBRNe)

inputs  $\alpha$  and  $\delta$  were observed to greatly affect the stability of Striker's movement and the constancy of the speed. The Sidewinding gait had four distinct poses that characterized it (Figure 4.2.1A).

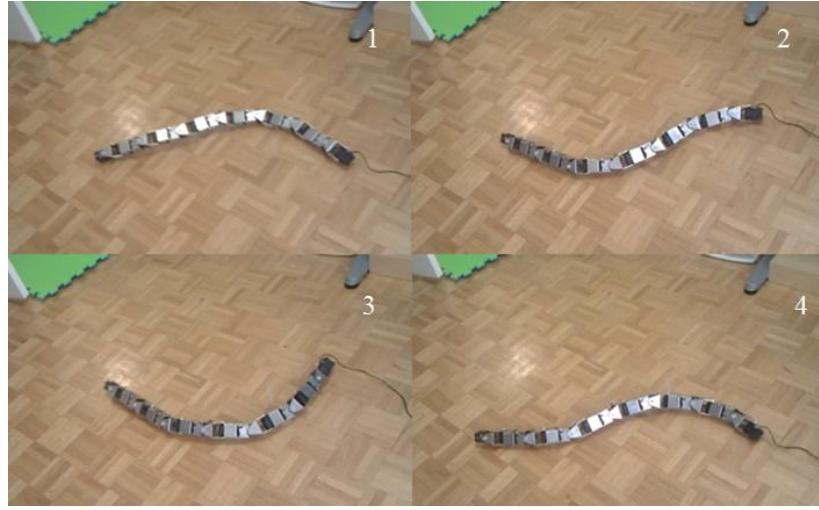


Figure 4.2.1A: Sidewinding gait in four distinct poses

Lateral Rolling was implemented using equation (7) from chapter 2 with the following input parameters value:  $\alpha = 0.5$ ,  $\delta = \pi/3$ ,  $s = 20$ ,  $K_n = 1$ ,  $n_\theta = 7$ ,  $n_\phi = 7$ ,  $l = 14$ . We observed Striker rolling sideways at approximately 4.7 metres/min. This was observed to be the fastest gait due to its simple composite motions of contracting and expanding in an arc. This motion allowed all segments to affectively push the entire body forward (upward in Figure 4.2.1B); thus, resulting in the fastest gait in the PMT.





Figure 4.2.1B: the lateral rolling gait pushed Striker on its side (upward) by contracting and expanding the arc.

#### 4.2.2 Experimental results for IMT:

Striker achieved 0.45 metres/minute on the IMT with the use of the concertina gait using equation (5). Input parameters for the equation were as follow:  $\alpha = 0.3$ ,  $s = 10$ ,  $K_n = 3$ ,  $n=7$ ,  $l=14$ . In this test it was observed that the wave height had to be low, on an  $18^\circ$  incline, which allowed for segments to achieve more traction as shown in Figure 4.2.2.



Figure 4.2.2: Striker scaling an  $18^0$  inclined wood surface using the concertina gait.

### 4.2.3 Experimental results for NPMT

The concertina gait (Equation 5 ) with the same input parameters,  $\alpha = 0.5$ ,  $s = 10$ ,  $K_n = 2$ ,  $K_1 = 0$ ,  $n=7$ ,  $l=14$ , allowed for a top speed of 1.6 metres/minute through a passage of 0.065m wide. Via the EGT method we created a simple two-pose gait dubbed the SG1 gait . The SG1 was able to move 0.67 metres/minute with a passage of 0.085 metres wide. Frame 1 of the SG1 gait is a straight pose with all joint angles associated to  $0^0$  or 511 actuator value. This pose is used for calibration before motion. The second pose is akin to a rope being twisted or a corkscrew (Figure 4.2.3). SG1's FRAME1 and FRAME2 joint angles are given in Table 4.2.3.



Figure 4.2.3: SG1 gait containing 2 Frames, Frame 1 and Frame 2 respectively.

Table 4.2.3: SG1 gait created with the EGT method.

Frame	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11	q12	q13	q14
F1	511	511	511	511	511	511	511	511	511	511	511	511	511	511
F2	567	631	647	319	311	717	726	286	295	681	689	363	357	575

This Gait was trained to exploit the side walls of the test. The twisting and straightening motion between the two frames pushes against the side wall for stability and allows for an increase in speed. The SG1 gait is an extremely quick and easy way to create mobility for narrow passages.

#### 4.2.4 Experimental results for OMT

Using wood we constructed a triangular prism shaped obstacle with the following dimensions: base of 39cm, two sides of 27cm wide, and 122cm length. The triangular prism simulates obstacles that have relatively low profile but have steep incline surfaces and have sharp vertex. There is nothing in the literature concerning a control law capable of traversing such a shape; thus, we trained one using the EGT method. The trained gait is composed of 20 Frames detailed in table 4.2.4, referred to as the “SG2” gait.

Table 4.2.4: The SG2 gait has 20 poses.

Frame	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11	q12	q13	q14
<b>F1</b>	511	511	511	511	511	511	511	511	511	511	511	511	511	514
<b>F2</b>	511	513	513	511	506	521	506	490	365	531	467	511	531	511
<b>F3</b>	511	513	519	487	503	542	471	534	339	583	471	606	530	635
<b>F4</b>	537	550	521	545	494	556	482	534	336	604	418	609	491	632
<b>F5</b>	536	541	524	509	497	567	403	592	389	612	476	643	511	728
<b>F6</b>	536	558	532	554	431	653	380	523	531	471	605	604	645	657
<b>F7</b>	536	661	501	573	386	679	470	438	543	429	626	507	641	656
<b>F8</b>	575	508	505	620	359	598	510	425	624	429	576	500	587	608
<b>F9</b>	574	509	415	656	441	533	548	429	624	547	615	527	587	609
<b>F10</b>	458	511	318	667	640	497	608	445	595	544	526	591	545	609
<b>F11</b>	464	455	481	444	657	506	594	487	523	668	529	515	525	609
<b>F12</b>	463	455	475	435	468	518	731	537	566	668	484	479	525	454
<b>F13</b>	433	458	492	528	507	503	758	404	487	471	479	384	526	454
<b>F14</b>	454	691	478	474	609	572	721	421	460	456	463	416	526	454
<b>F15</b>	520	536	503	465	664	662	619	605	458	478	456	380	529	639
<b>F16</b>	519	531	499	455	557	782	508	504	457	481	457	398	529	639
<b>F17</b>	685	466	497	167	406	614	453	485	455	470	456	398	529	639
<b>F18</b>	629	467	183	188	318	598	424	453	673	504	636	532	531	589
<b>F19</b>	628	466	320	506	323	495	416	467	675	499	634	542	531	589
<b>F20</b>	511	511	511	511	511	511	511	511	511	511	511	511	511	511

This unassuming obstacle turned out to be very challenging to overcome. Due to its smooth and slanted surfaces, Striker had nothing to grab hold of except for the vertex of the obstacle (Figure 4.2.3). With the use of the SG2 gait, Striker overcame the OMT in a very speedy 26 seconds.



Figure 4.2.4: Striker overcoming OMT using the SG2 gait.

Due to Striker's slender body it was evident that balance is important to keep the poses in place and to keep Striker stable while moving between poses. This is especially true when transitioning the front segments from the ground to the vertex of the the obstable shown in photo numbered 1 in Figure 4.2.4.

#### 4.2.5 Experimental results for SMT:

The SMT tested was constructed of wood and had 26 cm height, 24 cm width for the stair treads and 122cm length. A gait dubbed "SG3" was trained using the EGT method

especially for this test bed. The SG3 gait is composed of 24 Frames (Table 4.2.5) and Striker took 27 seconds to climb one flight of stairs.

Table 4.2.5: SG3 joins angles of 24 Frames

Frame	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11	q12	q13	q14
<b>F1</b>	511	511	511	511	511	511	511	511	511	511	511	511	511	511
<b>F2</b>	441	577	526	564	475	791	543	479	502	513	535	504	566	508
<b>F3</b>	477	681	287	575	485	749	445	631	506	486	581	516	566	507
<b>F4</b>	477	518	299	577	375	542	469	678	495	499	580	525	565	507
<b>F5</b>	493	252	334	577	391	611	470	668	523	482	581	516	566	506
<b>F6</b>	477	411	383	402	337	545	488	797	481	471	413	540	526	505
<b>F7</b>	368	397	422	353	318	484	356	703	491	720	453	508	515	457
<b>F8</b>	369	398	514	342	328	468	340	589	483	807	461	492	516	454
<b>F9</b>	417	399	502	345	385	305	314	561	468	688	381	678	510	477
<b>F10</b>	420	439	481	399	484	279	279	489	442	615	418	760	494	373
<b>F11</b>	469	359	503	406	574	403	460	283	325	475	368	659	568	688
<b>F12</b>	468	360	539	407	477	556	550	285	377	379	382	525	568	725
<b>F13</b>	466	485	528	456	473	573	570	319	382	514	281	483	568	530
<b>F14</b>	467	499	521	303	501	514	500	322	432	725	264	486	566	529
<b>F15</b>	467	466	476	302	472	528	509	402	424	717	266	484	568	528
<b>F16</b>	467	466	538	297	471	467	513	384	446	767	644	548	622	528
<b>F17</b>	467	466	498	298	472	468	494	385	667	822	606	591	622	528
<b>F18</b>	467	467	469	301	488	500	512	534	633	812	538	451	620	529
<b>F19</b>	467	500	499	455	508	512	500	496	525	789	530	451	620	529
<b>F20</b>	468	498	500	455	505	524	512	608	516	681	529	499	622	528
<b>F21</b>	469	499	501	456	590	561	428	490	434	604	613	672	582	527
<b>F22</b>	469	499	575	457	486	511	382	493	595	605	540	675	547	527
<b>F23</b>	469	498	521	496	502	504	494	493	530	606	529	675	547	527
<b>F24</b>	511	511	511	511	511	511	511	511	511	511	511	511	511	511

The SMT required Striker to transfer itself from a lower platform to a higher one. To accomplish this we trained Striker to have an initial pose that acted as a steady anchor on the higher surfaces as shown in photo numbered 2 of Figure4.2.5. The anchor helped relieved some of the stresses from the lower segment joints. The higher segments pulled as the lower segments simultaneously pushed to get more segments onto the tread of the



stair (photo 3 of Figure 4.2.5). This relationship between the two planes continued until at least half of the segments got onto the tread, at which point the focus was to pull the remaining segments up as shown in photos 3 through 5 of Figure 4.2.5.

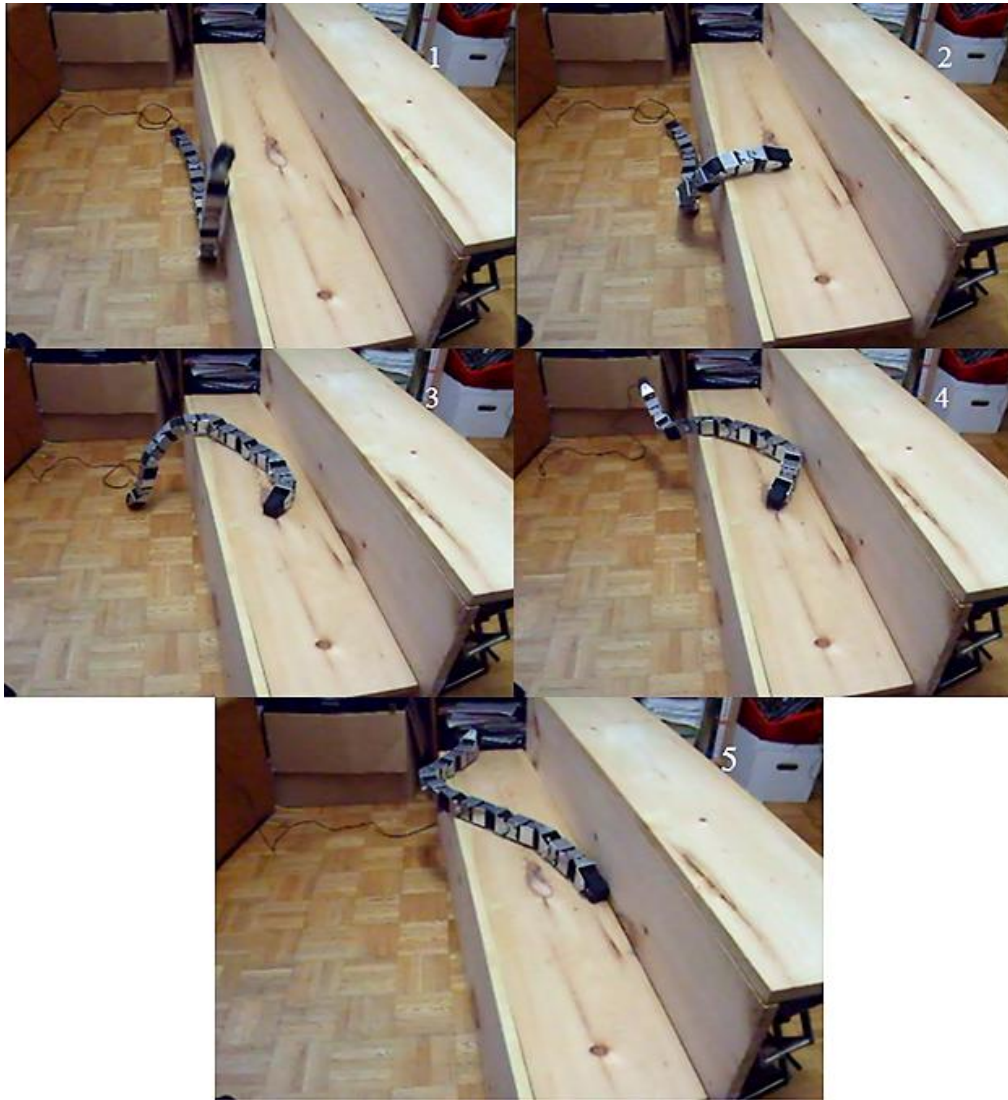


Figure 4.2.5: Striker climb stair using SG3

It was observed that this test really stressed Striker and its weakness was very apparent. The joint torque limit was the key factor for success in climbing the step, but at several

poses nearly half of Striker's body segments were being lifted and maintained by just one joint. This caused the actuators to overheat and fail. The joint angle limit is also a consideration because it limits the type of poses possible in the current situation.

### **4.3 The Challenge**

Dr. Alex Ferwon presented a video of a variable geometry tracked robot made by Engineering Services Inc. [71] attempting to scale an inclined slab of concrete on the rubble pile at the OPP UCBRNT<sup>6</sup> headquarters in Bolton, Ontario as depicted in Figure 4.3.1. This rubble pile is designed to simulate settings found in urban disasters mostly for the training of search dogs and rescue technicians.. The robot with its variable geometry tracks ultimately failed to properly climb the slab of inclined concrete in several attempts. The short and unsteady video clip was taken by hand at an unknown orientation; thus, it was difficult to gage the degree of the incline and the surface condition. Dr. Ferworn, then challenged Striker using the EGT method to attempt to scale the exact same surface that ESIT's robot had failed to climb.

---

<sup>6</sup> Urban Search and Rescue, Chemical, Biological, Radiological, Nuclear explosive Response Team (UCBRNT)





Figure 4.3.1: Concrete inclined surface at OPP UCBRNT training facility

In preparation for the challenge with an uncertain surface condition and inclined angle, Striker's metal skin was augmented to maximize contact friction. Assuming the worst case scenario, the first step in the preparation was to find the maximum incline angle which Striker could stay on a similar surface without sliding downward. We tested several silicon pads, a variety of plastic and rubber materials, leather and others on pine wood surfaces.

Leather was selected even though it was not the one that provided the best grip. Our decision to use leather was because of its durability and relatively low maintenance outweighed a negligible difference from the top performer (See Appendix D for the complete preparation process). We attach the leather to the underside (belly) of Striker (Figure 4.3.2).

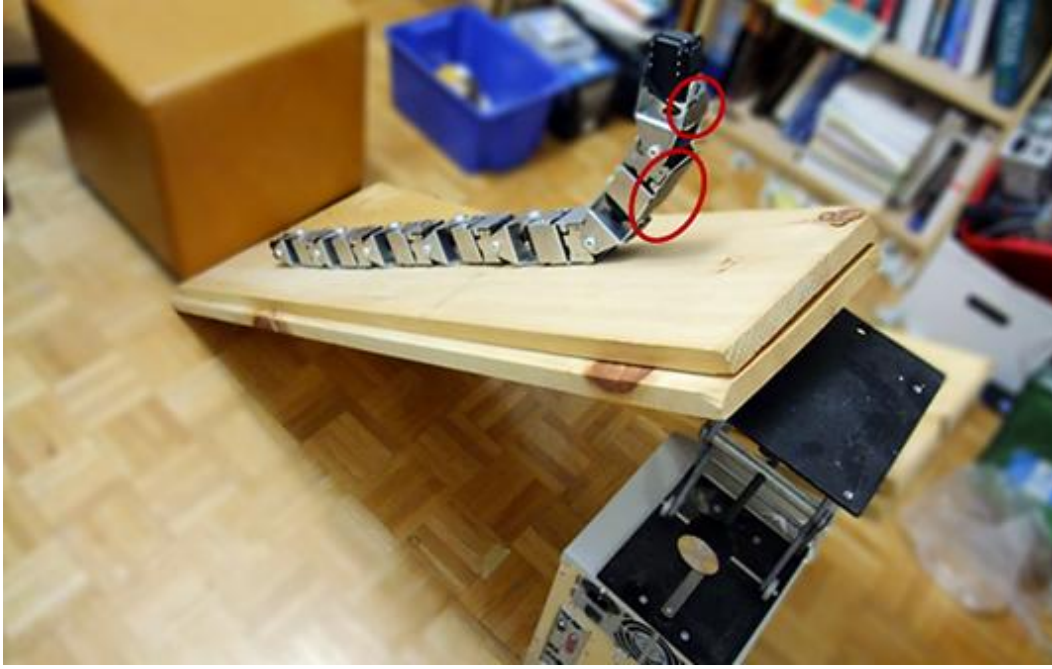


Figure 4.3.2: Striker on a slope with leather (red circles) augmentation attached to the belly.

After several attempts with different gaits we were succeed in creating a gait using the EGT method that enabled Striker to climb at an angle up to  $27^0$ . The gait is composed of 24 Frames and is dubbed “Hillz”. Hillz is a relatively simple gait which esentially repeats a serries of five frames along Striker’s body segments. For complete details of the Hillz gait Frames refer to Table 4.3.

Table 4.3: The Hillz Gait is compose of 24 FRAMES

Frame	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11	q12	q13	q14
<b>F1</b>	511	511	511	511	511	511	511	511	511	511	511	511	511	511
<b>F2</b>	511	757	522	436	498	503	510	497	520	496	511	510	509	577
<b>F3</b>	511	711	523	610	501	411	511	506	516	499	511	510	509	576
<b>F4</b>	511	558	523	741	503	355	511	506	516	499	511	510	509	576
<b>F5</b>	511	379	515	791	512	364	511	504	518	498	511	510	509	576

<b>F6</b>	511	357	514	739	508	524	511	413	518	501	511	510	509	576
<b>F7</b>	511	390	513	643	508	646	512	364	515	501	511	510	509	576
<b>F8</b>	511	387	513	542	507	721	511	345	515	501	509	504	510	574
<b>F9</b>	511	511	513	387	507	774	512	354	515	501	509	504	510	574
<b>F10</b>	511	509	519	360	485	744	512	498	516	425	511	505	510	575
<b>F11</b>	511	509	519	390	485	625	512	664	516	376	511	505	510	575
<b>F12</b>	511	522	519	442	486	522	513	727	518	364	511	505	510	575
<b>F13</b>	511	523	519	522	487	370	513	769	516	364	511	505	510	575
<b>F14</b>	511	523	519	521	487	351	513	736	518	496	511	426	510	575
<b>F15</b>	511	522	519	524	491	374	510	641	516	636	512	363	510	574
<b>F16</b>	511	522	519	524	491	425	513	511	516	729	511	345	510	574
<b>F17</b>	511	522	519	524	491	500	513	372	515	751	512	363	510	574
<b>F18</b>	511	522	519	524	491	501	513	343	516	743	511	492	510	433
<b>F19</b>	511	522	519	524	491	501	513	362	515	670	510	639	510	400
<b>F20</b>	511	522	519	522	491	503	513	413	515	551	510	733	510	352
<b>F21</b>	511	522	519	519	491	514	513	483	515	400	510	781	510	352
<b>F22</b>	511	522	519	520	488	513	512	484	518	402	510	708	510	216
<b>F23</b>	511	522	519	520	490	513	512	484	520	478	512	439	510	512
<b>F24</b>	511	522	519	520	490	513	512	484	520	494	511	505	510	452

Upon arrival at UCRT headquarters, we measured the incline angle to be approximately  $32^{\circ}$ . The surface was slightly textured and was not as smooth as the tested pine wood surface. Striker with its augmented skin was successful in climbing the incline surface using the Hillz gait( Figure 4.3.4). From Figure 4.3.4, F2 to F6 the depict the complete cycle of the sequence of motions for one sections. The sequences of motion following the first cycles are similare to {F7,F3,F4,F5,F6}.



Figure 4.3.4: Using the Hillz gait, Striker climbed an inclined concrete slab in a real settings.

The climb was successful but it did have a restriction; Striker had to be placed on the surface before the Hillz gait could begin.

#### 4.4 Analysis and Summary

Striker was able to negotiate all of the SMMSR tests with the use of various gaits from mathematical control laws and ones from the EGT method. The gaits using control laws from the literature were smooth and easy to control once they were properly tuned. To tune the serpentine, concertina, sidewinding and lateral rolling gait to the current state was difficult and took several weeks due to the sheer number of input variables and their dependencies. The positive side to the tuning process is that it allowed for greater insight into the mechanism of snake robot locomotion. The serpentine gait was not applicable to Striker since it only controlled movements parallel to the surface. The concertina gait

was capable of PMT, IMT and NPMT. Lateral rolling and Sidewinding were only applicable to the PMT. Both Concertina and Sidewinding gaits are greatly affected by how high the body segments were lifted during locomotion, but having them relatively flat to the surface will also affect their performance due to increased contact friction. Thus the amplitude of the wave must be carefully tuned. The first three tests were straight forward with continuous flat surfaces. These types of surfaces did not really stress Striker, as the results showed Striker was able to achieve relatively smooth locomotion at relatively high speed.

The EGT methodology was shown to be an effective tool for creating new kinds of locomotion in a very short period of time. The SG1 gait was created in less than five minutes using the EGT method. The SG1 gait was designed for the NPMT but was not able to achieve the same speed as the concertina gait. The concertina gait is completely dependent on the floor surface and therefore may not be as effective if the surface is not regular. The SG1 was designed to use both the floor surface and side walls to achieve forward motion.

The SG2 and SG3 gaits were also created using the EGT method but this time each took about two and three hours to create, respectively. The difficulties of the surfaces required each gait to have higher Frame resolution to properly make use of the obstacles' features. The OMT with sharp, short inclines presented a problem because we could not place the robot on the surface to start. The OMT required the trainer to consider anchor placement and leverage. The SG2 used Striker's latter section as an anchor to lift the front portion

into position and then hook onto the obstacle's vertex. Once hooked, the front segments helped pull the next few segments up. The anchored segments simultaneously pushed as the hooked segments pulled. The distributed efforts by the front and back sections of Striker gradually brought more segments onto the vertex of the obstacle and eventually overcame it. The same principle of distributed efforts was used to design the SG3 gait for the SMT. The SMT was more difficult because it only had right angle surfaces. The tread was flat and smooth and did not allow for a good grip and therefore the majority of the work was done by the section on the lower plane. All experimental results garnered on the SMMSR's surfaces are summarized by Table 4.4.

Table 4.4: Experimental results

	PMT (Planar)	IMT (Incline)	NPMT (Narrow)	OMT (Obstacle) 122x27x39	SMT (Step) 122x26x24cm
Serpentine					
Concertina	1.6 m/min	0.45m/min @ 18 <sup>0</sup>	1.6 m/min @ 5.7cm		
Lateral rolling	4.7m/min				
Sidewinding	1.66m/min				
SG1			0.67 m/min @ 8.5cm (2F)		
SG2				26sec (20F)	

SG3					27Sec (24F)
Hillz *		0.27m/min @27 <sup>0</sup> (24F)			

During training of new gaits with the EGT method we observed that there must be traction to allow forward propulsion. When the anchored segment(s) exhibit forces parallel to the current surface and in opposition to the intended direction, the effort is lost or minimized. This scenario may also create situations where slipping is a problem. The force created by the robot must be exerted onto the surface with an angle that maximizes traction to allow forward displacement.

The challenge presented by Dr. Ferworn caused a lot of turmoil because of the uncertainties in the terrain: The inclined angle and surface conditions were unknown. However, uncertainty in the terrain is a key factor that must be considered with mobile robotics. Without the EGT method to create new gaits, Striker would not have been able to climb the concrete incline on the rubble pile. The challenge resulted in more confidence in the EGT method and Striker's ability to overcome obstacles.

## **4.5 Summary**

Through the SMMSR tests and the real world challenge, Striker has been shown to be able to traverse several types of terrain. This experiment showed that a good test-bed is an effective tool to gauge similar real world settings. The EGT method allowed the trainer to create new gaits on-the-fly that enabled Striker to travel on terrains that it would not have been able to traverse using other methods.





## Chapter 5

# Conclusion and Future Work

### 5.1 Conclusion

The results presented in this thesis demonstrate that the use of the Explicit Gait Training method can provide snake robots, such as Striker, the ability to function in near real world settings, and the Standard Mobility Metric for Snake Robots is an effective test flat-form to evaluate snake robot locomotion. Snake robots have great potential to greatly improve the mobility of ground-based robots, but this potential is useless if it cannot be demonstrated.

The Explicit Gait Training method has been shown to work and we argue is an elegant solution that allows the potential of snake robots to be realized. The method allows new gaits to be created on-the-fly using the snake robot itself as a programming interface. The EGT method provides robot operators a fast tool for enhancing mobility by making new gaits quickly. This was not possible before our method.

The EGT method is not simply elegant but it is also flexible and has a great number of potential applications; it allows snake robots to affectively deal with uncertain terrain such as disaster sites, inspection in confined and narrow spaces or stuck situations. Striker, using the EGT method, showed that disjointed terrains and obstacles can be

overcome with relative ease. At the very least, EGT provides the only means by which unexpected terrain may be dealt with successfully.

EGT is not a miracle and we are not claiming that it supplants known snake gaits. When using the EGT method to create new gaits, one must take into consideration the joint torque load, leverage and anchoring. Leverage and anchoring are fundamental to exploit terrain features for locomotion. The largest limitation of the EGT method is the human factor. While anyone can use the EGT method to create new gaits, only people with the knowledge and ability to express those knowledge to successfully exploit the terrain will be able to create effective gaits. Hence effectiveness of the EGT method is directly dependent on the person creating the gait. The EGT method is not meant to replace existing snake gaits which are very effective in known terrain as presented in the literature, but the EGT method can be used in conjunction with existing gaits and complements them.

Striker is a capable snake-like machine with potential to traverse many types of terrain. Striker's ability to be controlled by traditional mathematical models as well as the EGT method demonstrated some of the snake robot's flexibility. Modularly designed, Striker is robust to mechanical failure. During the course of experimentation several actuating links failed but Striker was able to continue to locomote. The failed links were all quickly replaced thanks to the modular design of the segments. Modularity also gives Striker the ability to be lengthened or shortened as needed. Striker's metal skin did a good job of providing structural protection of its innards but it can be improve upon. The

aluminum housing is not as light as originally thought and provides only adequate traction. The limitation of the contact grip provided aluminum housing reduces the effectiveness of Striker's locomotion. The Augmenting of the metal skin with leather vastly improved Striker's grip. This was demonstrated in the challenge. Another limitation of Striker is the tethered power source. The length of tethered limits the distance Striker can travel. The tether may also be snag by obstacles and sharp edges.

The Standard Mobility Metric for Snake Robot provided good test bed to evaluate Striker's ability to locomote. Using the SMMSR we were able to simulate conditions of the concrete incline on the rubble to prepare Striker for the actual climb. Currently, the full potential of SMMSR not realized by the fact that only one snake robot had been evaluated, but this will change as more snake robot s get evaluated on this test flat form.

Striker and the EGT method showed that snake robots can be successfully used on terrain that cannot be traversed by other type of mobile robots and demonstrated that it is possible to create gaits quickly, apply them in unstructured environments and to overcome obstacles and other impediments to motion relatively quickly. A glimpse of this potential was demonstrated when Striker scaled the concrete incline on the rubble pile using gait created via the EGT method. Ultimately, the contributions of this work have brought snake locomotion closer to becoming a practical tool in a potentially wide variety of ground environments, not the least of which are those typically present in urban disasters.

## 5.2 Future Work

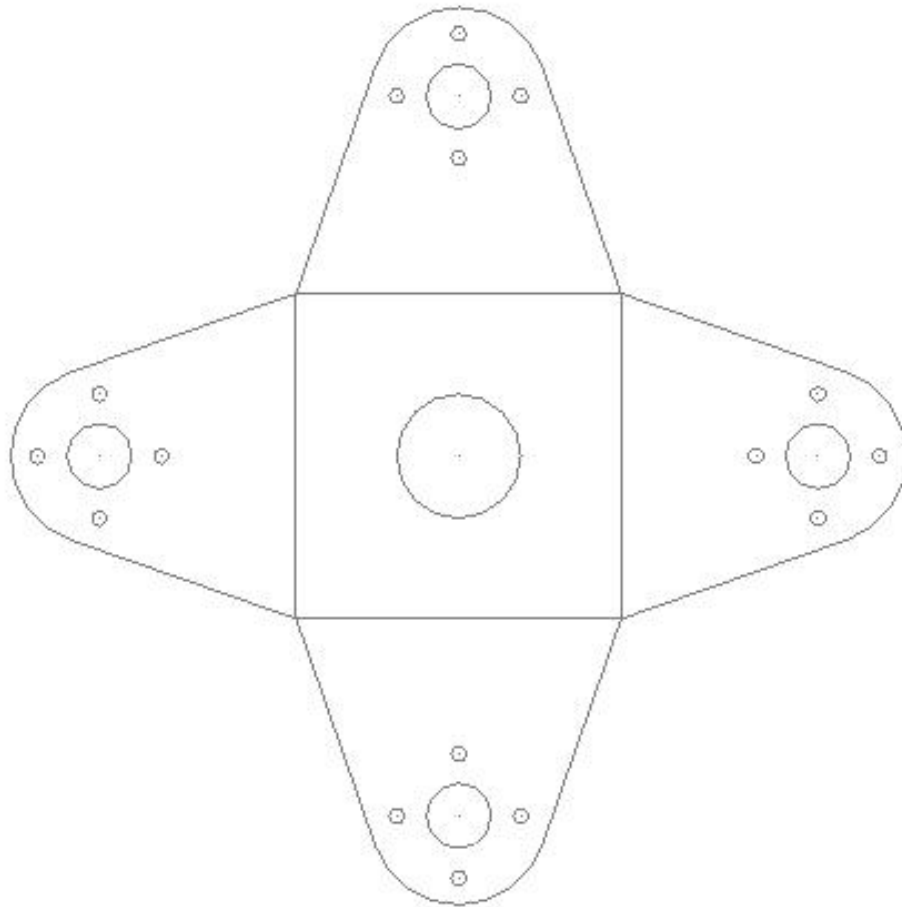
Further research will involve creating more gaits with the use of the EGT method on various terrains to gain more knowledge and experience with snake locomotion with the goal of determining general principles that can be applied to gait creation. This knowledge and experience may be used to create a new generation of gaits to resolve different locomotion problems that have not been practical until now.

Striker's skin may be changed to another material that would improve traction and reduce the overall weight. Currently, Striker provides no sensory information pertaining to the contact between its links and the surface. A skin capable of contact force feedback would provide information on the effectiveness of the gait that can be applied to, in turn, improve the gait. A skin with force feedback sensors will allow more in-depth understanding of snake-like locomotion as well as other forms of locomotion.

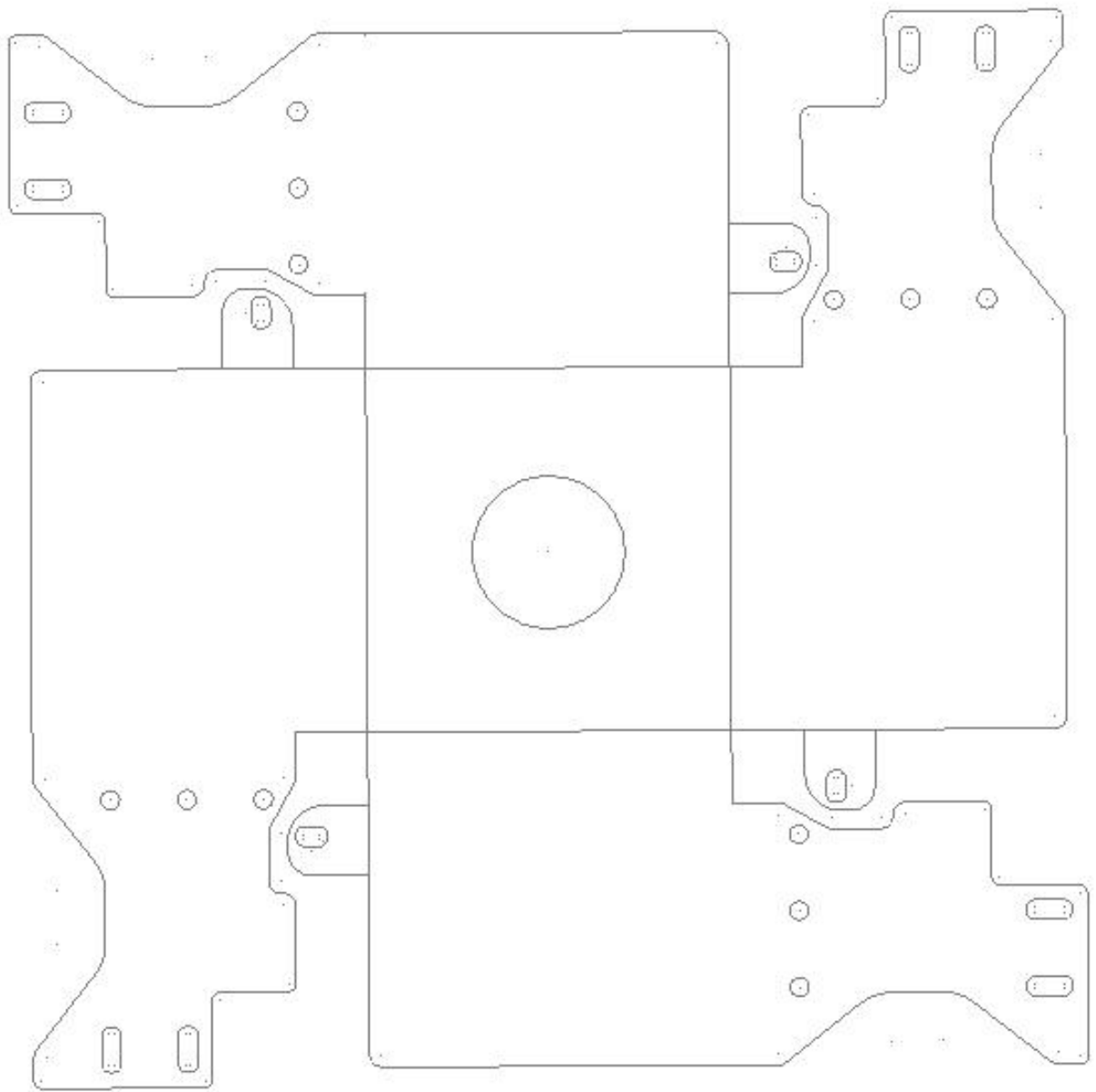
Ideally, other snake-like robots must be constructed to confirm that it is possible to use the methods suggested in this work in the general case. While this is beyond the scope of this work, we anticipate that others will make use of our work, confirm it and improve the locomotion of their snakes.

## Appendix A:

Unfold CAD designs for the Joint and segment housing brackets:



Unfolded joint bracket



Unfolded segment housing

## Appendix B:

Packet Composition Library to enable communication between the smart actuators and the NSR microcontroller.

```
/*InitialcAL)
```

Calibration function; it essentially straight out the Striker

```
*/
```

```
void InitialCal(int lower, int upper)
```

```
{
```

```
for(int i=lower; i<=upper;i++)
```

```
{
```

```
RegGoalPosition(i,512,100);
```

```
SendtoAX();
```

```
wait(10);
```



```
}
```

```
SendActionAX();
```

```
wait(200);
```

```
}
```

```
/*SendtoAx()
```

```
@param: use global array packet[]
```

```
@Function: send byte array to serial
```

```
*/
```

```
void SendtoAX()
```

```
{
```

```
int size;
```

```
size=packet[0];
```

```
for(int i=1; i<=size; i++)
```

```

{

Serial.print(packet[i],BYTE);

}

}

/*SendActionAx()

@param:

@Function: use together with reg_goalposition ,

send byte array to serial

*/

void SendActionAX()

{

byte action[]={6,255,255,254,2,5,250};

int size=action[0];

for(int i; i<=size; i++)

{

```

```

    Serial.print(action[i],BYTE);

}

}

byte F2Bytes[2];

/* get the load on the servo

Param: AX byte(id)

Return: complete package read to send to servo

*/

void getLoad(byte id)

{

    byte Instruction = 2;  //send the instruction to read

    byte AddressParam = 36;  //address of current load

    byte byteparam = 1; //number of byte(s) to read

    byte PackParam[3];

    PackParam[0] =2;

```

```

    PackParam[1] = AddressParam;

    PackParam[2] = byteparam;

    FormPacket(id, Instruction, PackParam);

}

/* get the position of servo

Param: AX byte(id)

Return: complete package read to send to servo    */

void getPosition(byte id)

{

    byte Instruction = 2;  //send the instruction to read

    byte AddressParam = 36;  //address of current position

    byte byteparam = 2; //number of byte(s) to read

    byte PackParam[3];

    PackParam[0]=2;

    PackParam[1] = AddressParam;

```

```

    PackParam[2] = byteparam;

    FormPacket(id, Instruction, PackParam);

}

/* get the temperature of servo

Param: AX byte(id)

Return: complete package read to send to servo    */

byte getTemperature(byte id){

    byte Instruction = 2;  //send the instruction to read

    byte AddressParam = 43;  //address of current temperature

    byte byteparam = 1; //number of byte(s) to read

    byte PackParam[3];

    PackParam[0]=2;

    PackParam[1] = AddressParam;

    PackParam[2] = byteparam;

    FormPacket(id, Instruction, PackParam);

```

```
}
```

```
/* get the voltage of servo
```

```
Param: AX byte(id)
```

```
Return: complete package read to send to servo    */
```

```
void getVoltage(byte id)
```

```
{
```

```
    byte Instruction = 2;  //send the instruction to read
```

```
    byte AddressParam = 42;  //address of current voltage
```

```
    byte bytewidth = 1; //number of byte(s) to read
```

```
    byte PackParam[3];
```

```
    PackParam[0]=2;
```

```
    PackParam[1] = AddressParam;
```

```
    PackParam[2] = bytewidth;
```

```
    FormPacket(id, Instruction, PackParam);
```

```
}
```

```
/*
```

Write

Param: AX byte(id), address in servo to write , value to write

Return: complete packet read to send to servo\*/

```
void Write(byte id, byte Insaddress, int value)
```

```
{
```

```
    byte InsWrite = 3;
```

```
    int aSize = 0;
```

```
    byte PackParam[4];
```

```
    if (value > 255)
```

```
    {
```

```
        aSize = 3;
```

```
    }
```

```

else

{

    aSize = 2;

}

PackParam[0]=aSize;

PackParam[1] = Insaddress;

if (value > 255)

{

    Format2Bytes(value);

    PackParam[2] = F2Bytes[0];

    PackParam[3] = F2Bytes[1];

}

else

{

    PackParam[2] = (byte)(value);

}

```



```

        FormPacket(id, InsWrite, PackParam);

    }

/* !n02122009

Reg_GoalPosition Using reg write instruction

!need action instruction after

Param: AX byte(id), goal position, moving speed

Return: complete package read to send to servo

*/

void RegGoalPosition(byte id, int position, int speed)

{

    byte GoInstruction = 4;

    byte AddressParam = 30;

    byte Temp2bytes[2];

    byte PackParam[6];

    PackParam[0] = 5;

```

```

    PackParam[1] = AddressParam;

    Format2Bytes(position);

    PackParam[2] = F2Bytes[0];

    PackParam[3] = F2Bytes[1];

    Format2Bytes(speed);

    PackParam[4] = F2Bytes[0];

    PackParam[5] = F2Bytes[1];


    FormPacket(id, GoInstruction, PackParam);

}

/* !n10182008

GoalPosition

Param: AX byte(id), goal position, moving speed

Return: complete package read to send to servo

*/

```

```

void GoalPosition(byte id, int position, int speed)

{

    byte GoInstruction = 3;

    byte AddressParam = 30;

    byte Temp2bytes[2];

    byte PackParam[6];

    PackParam[0] = 5;

    PackParam[1] = AddressParam;

    Format2Bytes(position);

    PackParam[2] = F2Bytes[0];

    PackParam[3] = F2Bytes[1];


    Format2Bytes(speed);

    PackParam[4] = F2Bytes[0];

    PackParam[5] = F2Bytes[1];

    FormPacket(id, GoInstruction, PackParam);}

```

```
/*=====supporting functions==10172008=====*/
```

```
/* creates packet
```

```
@Param: AX byte(id), goal position, moving speed
```

```
@Return: complete packet read to send to servo
```

```
*/
```

```
void FormPacket(byte id, byte instruction, byte parameters[])
```

```
{
```

```
    int TempChecksum = 0;
```

```
    byte Checkval = 0;
```

```
    int paramSize=parameters[0];
```

```
    byte paramLength = parameters[0] + 2;
```

```
    byte packetSize = parameters[0] + 7; //7=6+1(extra index use for packet
```

```
    packet[0] = packetSize -1; //sudo array size
```

```
    packet[1] = byte(255); //packet header
```

```
    packet[2] = byte(255); //"
```

```

packet[3] = id;

packet[4] = paramLength;

packet[5] = instruction;

for (int i = 1; i <=paramSize; i++)

{

    packet[5 + i] = parameters[i];

    TempChecksum += (int)parameters[i];

}


TempChecksum += id + paramLength + instruction;

Checkval = CheckSum(TempChecksum);

packet[packetSize - 1] = Checkval;

// return packet;

}

/* formats a 2 byte array

```

@Param: Takes value that require byte spaces ie. 200 or 500

@Return: format value 2 bytes array \*/

```
void Format2Bytes(int value)
```

```
{
```

```
    int firstval = 0;
```

```
    int secondval = 0;
```

```
    if (value > 255)
```

```
    {
```

```
        firstval = value % 256; // get the remainder
```

```
        secondval = value / 256;
```

```
        F2Bytes[0] = (byte)firstval;
```

```
        F2Bytes[1] = (byte)secondval;
```

```
    }
```

```
    else
```

```
    {
```

```
        F2Bytes[0] = (byte)value;
```

```

        F2Bytes[1] = 0;

    }

}

byte CheckSum(int checksumval)

{

    int cs;

    cs=~checksumval&0xff;

    return byte(cs);

}

void wait(int ms) //alternative to delay when using interrupt

{

    for( int i=0; i<ms; i++)

    {

        delayMicroseconds(1000);

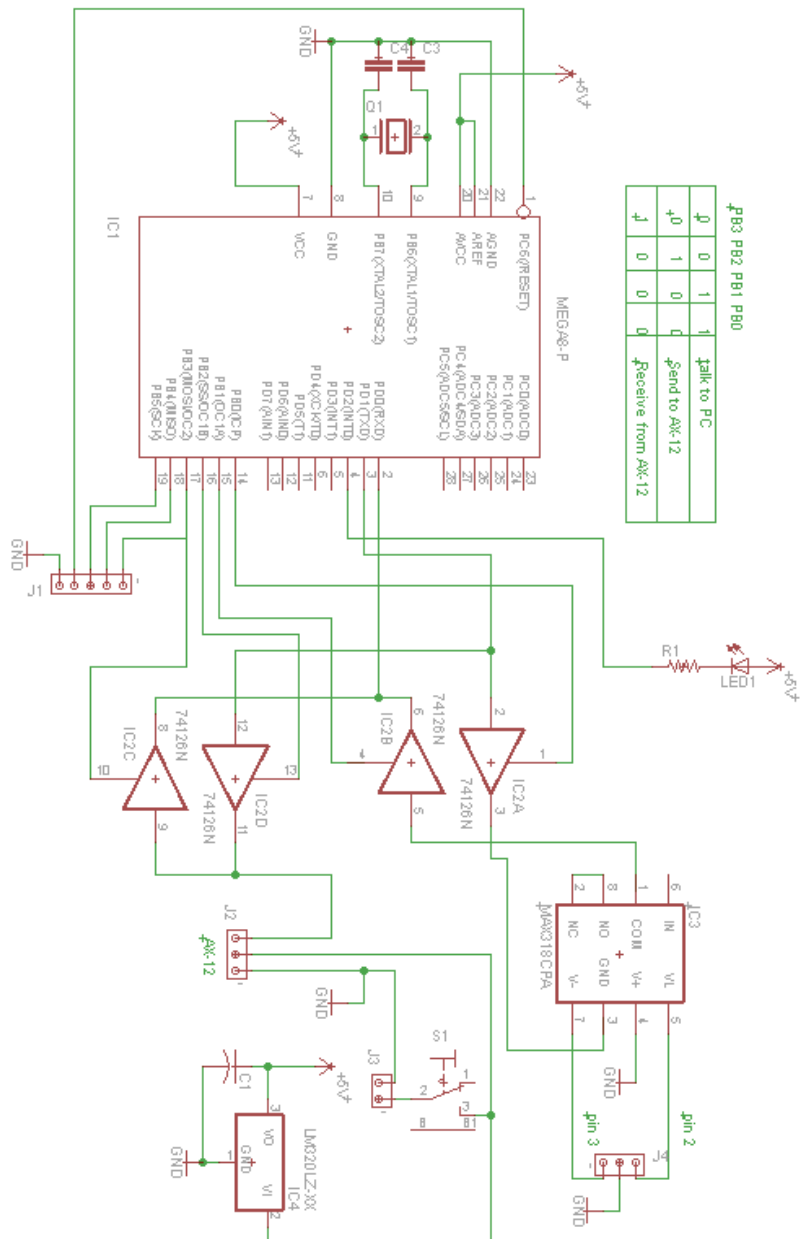
    }

}

```

## Appendix C

NSR microcontroller schematic for Striker





## Appendix D

### Preparation for the Challenge

Assuming the worst case scenario, the first step in the preparation was to find the maximum incline angle which Striker could stay on a similar surface without sliding downward. We tested several silicon pads, a variety of plastic and rubber materials, leather and others on pine wood surfaces. The ones we found worked best were two kinds of rubber shelf liners commonly found at department stores and leather as shown in Figure D1.

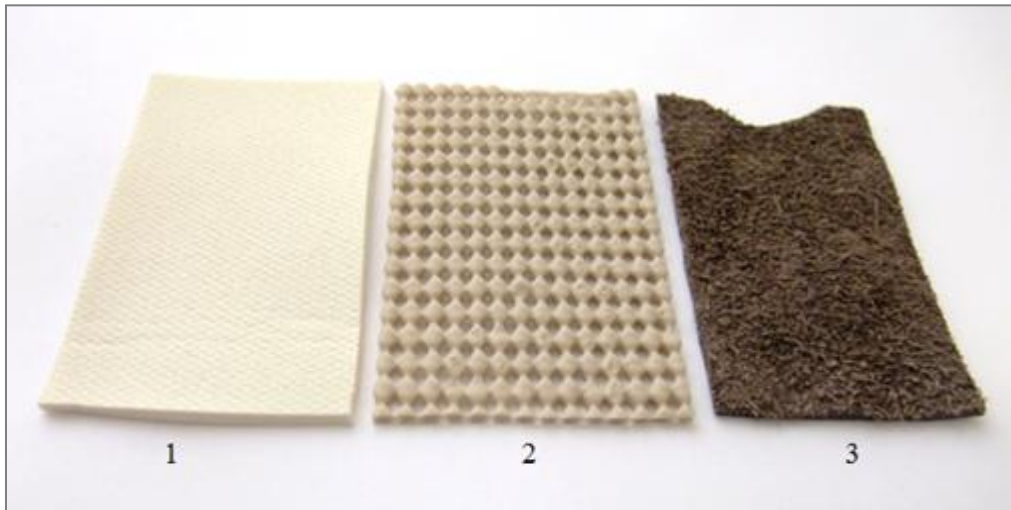


Figure D1: Metal skin augment materials: White shelf liner, perforated liner, and leather.

The three materials were very similar in providing a good grip with a variation of plus or minus 2 degrees of each other. The white liner (1) was able to maintained a grip at 41 degrees, while the back, rough, side of leather (3) was able to hold at 43 degrees, and the

perforated liner (2) performed well up to 45 degrees. Leather was selected even though it was not the one that provided the best grip. Our decision to use leather was because of its durability and relatively low maintenance outweighed a negligible two degrees difference from the top performer. We attach the leather to the underside (belly) of Striker using adhesive (Figure 4.4.3).

Using the EGT method to create a gait that would scale a  $40^{\circ}$  incline proved fruitless, as suspected; the frictional grip is reduced when Striker is in motion. After many attempts with different gaits we reduced the angle to  $27^{\circ}$  and Striker was able to climb using a 24 Frames gait dubbed “Hillz”.

## Bibliography

- [1] K. Byl, R. “Dynamically Diverse Legged Locomotion for Rought Terrain”. Proc. IEEE Robotics and Automation, pp. 1607-1608. Kobe, Japan, May 2009.
- [2] H. Yamamoto, K. Ohnishi. “An approach to stable walking on unknown slippery floor for biped robot”. Industrial Electronics Society, pp. 1728-1733. 2001. IECON '01. Denver, USA, Nov. 2001.
- [3] E. Celaya, J.M Porta. “New Trends in Legged Robots telecooperation”. IEEE Robot and Human Communication, pp. 232-237, Tsukaba, November 1996.
- [4] S. Nabulsi, J.F. Sarria, H. Montes, M.A.Armada. “High-Resolution Indirect Feet–Ground Interaction Measurement for Hydraulic-Legged Robots”. IEEE Instrumentation and Measurement, pp.3396-3404, Oct. 2009.
- [5] B.Goodwine, J. Burdick. “Gait controllability for legged robots”. IEEE Robotics and Automation, pp. 484-489, 1998. Leuven, May 1998
- [6] M. Heinen, F. Osorio. “Morphology and Gait Control Evolution of Legged Robots”. IEEE Robotic Symposium, LARS '08, pp. 111-116, October, 2008.
- [7] R. Chatterjee, M. Nagai, F. Matsuno. “Development of modular legged robots: study with three-legged robot modularity”. Intelligent Robots and Systems, (IROS 2004) pp. 1450-1455, Oct 2004.
- [8] F. Zonfrilli, G. Oriolo, D. Nardi. “A biped locomotion strategy for the quadruped robot Sony ERS-210”. Robotics and Automation, pp. 2768-2774 . Washington, DC May 2002

- [9] “<http://www.thecanadianencyclopedia.com/index.cfm?PgNm=TCE&Params=A1ARTA0008280>”. Aug. 2010.
- [10] Population Density  
“<http://www.un.org/esa/population/publications/WUP2005/2005wup.htm>”, Aug 2010.
- [11] World Urbanization Prospects 2009 Revision. United Nation website:  
“<http://esa.un.org/unpd/wup/index.htm>”, Aug. 2010.
- [12] D. Cuberes. “Sequential city growth: Empirical evidence”. *Journal of Urban Economics* 69 (2), pp. 229-239, 2011.
- [13] S. Poelhekke. “Urban Growth and Uninsured Rural Risk: Booming towns in bust times”. *Journal of Development Economics* Jul. 2010
- [14] Craig J.J. “ Introduction to Robotics: Mechanics and Control”, 2<sup>nd</sup> ed., pp. Addison-Wesley, Mass 1989
- [15] S. Hirose, “Active code mechanism no.3"ACM III",” 1973. [Online]. Available: [http://www.robot.mes.titech.ac.jp/robot/snake/acm3/acm3\\_e.html](http://www.robot.mes.titech.ac.jp/robot/snake/acm3/acm3_e.html). Jan. 2009.
- [16] S. Hirose and Y. Umetani, “Kinematic control of active cord-mechanism with tactile sensors,” in *Proc. 2nd RoMANSy Symp.*, Warsaw, 1976, pp. 241–252.
- [17] G. S. Chirikjian and J. W. Burdick, “Design and experiments with a 30 degree-of-freedom robot,” in *Proc. IEEE Int. Conf Robotics and Automation*, Atlanta, GA, May, 2-5, 1993, vol. 3, pp. 113-119

- [18] G. S. Chirikjian and J. W. Burdick, "An obstacle avoidance algorithm for hyper-redundant manipulators," in *Proc. IEEE Int. Conf. Robotics and Automation*, Cincinnati, OH, May 14-17, 1990, pp. 625-631.
- [19] Ostrowski, J. and J. Burdick (1996). "Gait kinematics for a serpentine robot". In: *Proc. IEEE Int. Conf. Robotics and Automation*. Vol. 2. pp. 1294-1299.
- [20] J. Gray, "The mechanism of locomotion in snakes," *J. Exp. Biol.*, vol. 23, no. 2, pp. 101-120, 1946.
- [21] Ye, C., S. Ma, B. Li and Y. Wang (2004b). "Twist-related locomotion of a 3D snake-like robot". In: *Proc. IEEE Int. Conf. Robotics and Biomimetics*. pp. 589-594
- [22] E. Shammass, A. Wolf, and H. Choset, "Journal of Mechanism and Machine Theory, April, 2005, pp. 170 – 190
- [23] Hidetaka Ohno, Shigeo Hirose : "Study on Slime Robot (Proposal of Slime Robot and Design of Slim Slime Robot)", *Proc IROS 00 Kagawa*, pp.2218-2223 (2000)
- [24] Liljebäck, P., Ø. Stavdahl and K. Y. Pettersen . "Modular pneumatic snake robot: 3D modelling, implementation and control". In: *Proc. 16<sup>th</sup> IFAC World Congress*. 2005
- [25] United Nations International Search and Rescue Advisory Group  
"http://ochaonline.un.org/OCHAHome/AboutUs/Coordination/INSARAG/tabid/6016/language/en-US/Default.aspx", Aug.2010
- [26] S. Tadokoro, H. Kitano, T. Takahashi, I. Noda , H. Matsubara, A. Shinjoh T. Koto, I. Takeuchi, H. Takahashi, F. Matsuno, Hatayama, J. Nobe, S. Shimada.

- “The RoboCup-Rescue project: a robotic approach to the disaster mitigation problem”. IEEE Robotics and Automation. San Francisco, pp. 4089-4094, CA May 2000.
- [27] A. Jacoff, E. Messina, B. Weiss, S. Tadokoro, Y. Nakagawa, "Test Arenas and Performance Metrics for Urban Search and Rescue Robots", IEEE/RSJ International Conference on Intelligent Robots and systems, pp.3396-3403, 2003.
  - [28] B.M. Hudock, , B.E. Bishop, F.L. Crabbe. "On the development of a novel urban search and rescue robot," System Theory, 2004. Proceedings of the Thirty-Sixth Southeastern Symposium on , vol., no., pp. 451- 455, 2004
  - [29] Murphy, R.; Casper, J.; Hyams, J.; Micire, M.; Minten, B.; , "Mobility and sensing demands in USAR," Industrial Electronics Society, 2000. IECON 2000. 26th Annual Conference of the IEEE , vol.1, no., pp.138-142 vol.1, 2000.
  - [30] R. Murphy, “Activities of the Rescue Robots at the World Trade Center from 11–21 September 2001,” IEEE Robotics & Automation Magazine, pp.50-61, 2004.
  - [31] S.Tadokoro, R. Verhoeven, M. Hiller, T. Takamori, "A portable parallel manipulator for search and rescue at large-scale urban earthquakes and an identification algorithm for the installation in unstructured environments," Intelligent Robots and Systems, 1999. IROS '99. Proceedings. 1999 IEEE/RSJ International Conference on , pp.1222-1227 vol.2, 1999
  - [32] S.A.A. Moosavian, H. Semsarilar, A. Kalantari. "Design and Manufacturing of a Mobile Rescue Robot" Intelligent Robots and Systems, 2006 IEEE/RSJ, pp.3982-3987, Oct. 2006

- [33] E. Messina, A. Jacoff, A. Virts. Response Robot: DHS/NIST Sponsored Evaluation Exercises -pocket guide. National Institute of Standards and Technology 2008.
- [34] Gans, C. Biomechanics Approach to Vertebrate Biology, [Reissue]. The University of Michigan Press, Ann Arbor, 1980.
- [35] S. Bennet, T. McConnel, and S. L Trubatch. "Quantitative analysis of the speed of snakes as a function of peg spacing". Journal of Experimental Biology, 60:161-165, 1975.
- [36] J.W. Burdick, J. Radford, and G.S Chirkjian. " A 'sidewinding' locomotion gait for hyper-redundant robots". Advance Robotics, pp.195-216. 1995.
- [37] Hirose, S. "Biologically Inspired Robots: Snake-like Locomotors and Manipulators". Ph.D thesis, Oxford University Press: Oxford 1993.
- [38] Hidetaka Ohno, Shigeo Hirose "Study on Slime Robot (Proposal of Slime Robot and Design of Slim Slime Robot)", Proc IROS 00 Kagawa, pp.2218-2223 (2000)
- [39] Liljebäck, P., Ø. Stavdahl and K. Y. Pettersen . "Modular pneumatic snake robot: 3D modelling, implementation and control". In: Proc. 16<sup>th</sup> IFAC World Congress, pp.450-453. 2005.
- [40] L. Sciavicco , B. Siciliano. "Modeling and Control of Robot Manipulators: 2<sup>nd</sup> " Springer publishing 2000, ISBN 1-85233-221-1.
- [41] K. Fu, R. Gonzales, C. Lee. "Robotics: Control, Sensing, Vision, and Intelligence" McGraw-Hill Book Company, Mass 1987.

- [42] Liljebäck, P., Ø. Stavdahl and K. Y. Pettersen .” Modular pneumatic snake robot: 3D modelling, implementation and control”. In: Proc. 16<sup>th</sup> IFAC World Congress. 2005
- [43] G. S. Chirikjian and J. W. Burdick, “Design and experiments with a 30 degree-of-freedom robot” in *Proc. IEEE Int. Conf Robotics and Automation*, pp. 113-119 Atlanta, GA, May, 1993.
- [44] G. S. Chirikjian and J. W. Burdick, “An obstacle avoidance algorithm for hyper-redundant manipulators,” in *Proc. IEEE Int. Conf Robotics and Automation*, pp.625-633, Cincinnati, OH, May, 1990.
- [45] G. S. Chirikjian and J. W. Burdick, “A modal approach to the kinematics of hyper-redundant manipulators,” *IEEE Trans. Robot. Automat.*, vol. 10, no. 3, pp. 343-354, June 1994.
- [46] G. Miller, “Neurotechnology for Biomimetic Robots. Chap. Snake Robots for Search and Rescue” pp. 271-284. MIT Press Cambridge, MA, USA. Cambridge/London. 2002
- [47] Ye, C., S. Ma, B. Li and Y.Wang . “Twist-related locomotion of a 3D snake-like robot”. *IEEE Int. Conf. Robotics and Biomimetics*, pp. 589-594. Oct, 2004b
- [48] Mori, M. and S. Hirose, “Three-dimensional serpentine motion and lateral rolling by active cord mechanism ACM-R3” *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*. pp. 829-834. Dec, 2002.



- [49] Togawa, K., M. Mori and S. Hirose. "Study on three-dimension active cord mechanism: Development of ACM-R2". In: Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems. Vol. 3. pp. 2242-2247. Takamatsu , Nov, 2000.
- [50] Yamada, H., S. Chigisaki, M. Mori, K.Takita, K. Ogami and S. Hirose. "Development of amphibious snake-like robot ACM-R5". In: Proc,36th Int. Symp. Robotics 2005.
- [51] Kevin J.Dowling. "Limbless Locomtion". Ph.D thesis, Carnegie Mellon University,1997.
- [52] Ohno, H. and S. Hirose. "Design of slim slime robot and its gait of locomotion." In: Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems. Vol. 2. pp. 707-715. Maui, HI, Aug, 2001.
- [53] S. Yu, S. Ma, B. Li; Y. Wang, "An amphibious snake-like robot: Design and motion experiments on ground and in water," Information and Automation, 2009, pp.500-505. June 2009.
- [54] K. Lipkin, I. Brown, A. Peck, H. Choset, J. Rembisz, P. Gianfortoni, A. Naaktgeboren "Differentiable and Piecewise Differentiable Gaits for Snake Robots" IEEE/RSJ Intl. Conf. Intelligent Robots and Systems, pp. 1864-1869, CA, USA, Oct 29 - Nov 2, 2007.
- [55] Israeli military's snake robot in the news: "<http://www.jewishjournal.com/chaitech/P16/>" Aug. 2009.

- [56] R. Worst, R. Linnemann, "Construction and operation of a snake-like robot," *Intelligence and Systems, 1996., IEEE International Joint Symposia on* , vol., no., pp.164-169, 4-5 Nov 1996
- [57] R. Linnemann, K. Paap, B. Klaassen, J. Vollmer, "Motion control of a snakelike robot," *Advanced Mobile Robots (Eurobot), 1999. Third European Workshop on* , vol., no., pp.1-8, 1999
- [58] B. Klaassen, K. Paap, "GMD-SNAKE2: a snake-like robot driven by wheels and a method for motion control," *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on* , vol.4, no., pp.3014-3019 vol.4, 1999
- [59] Liljebäck, P., Ø. Stavdahl and K. Y. Pettersen . "Modular pneumatic snake robot: 3D modelling, implementation and control". In: *Proc. 16<sup>th</sup> IFAC World Congress*. 2005
- [60] A. Transeth, R. Leine, C. Glocker, K. Pettersen, P. Liljebäck, "Snake Robot Obstacle-Aided Locomotion: Modeling, Simulations, and Experiments," *Robotics, IEEE Transactions on* , vol.24, no.1, pp.88-104, Feb. 2008
- [61] S. Ma, H. Araya, L. Li, "Development of a Creeping Locomotion Snake Robot", *International Journal of Robotics and Automation*, Vol. 17, pp. 146-153. No. 4, 2002.
- [62] Z. Li, S. Ma, B. Li, Y. Wang, "Serpentine Locomotion of a Snake-like Robot Controlled by Musical Theory". *Intelligent Robots and Systems(IROS)*, pp. 102-107. 2005.

- [63] B. Li, D. Tan, Y. Wang, L. Chen, S. Ma, "Study on a snake-like robot adapting to the ground," Intelligent Control and Automation ( WCICA 2004), vol.6, pp. 4877-4880 Vol.6, 15-19 June 2004
- [64] L. Chen, Y. Wang, S. Ma, B. Li; , "Studies on lateral rolling locomotion of a snake robot," IEEE Robotics and Automation, 2004. Proceedings. ICRA '04., vol.5,pp. 5070- 5074 , 26 April-1 May 2004
- [65] D. Barrett, T. King.“ Computer Networking Illuminated” 1<sup>st</sup> Ed., Jones and Bartlett Publishers, Inc. April 25, 2005. P92-200
- [66] G. Bancroft, F. Merritt. “3-D graphics applications in fluid flow simulations Computer Workstations”. IEEE Conference on Digital Object Identifier, pp. 142 – 147. 1988
- [67] K. Osuka , "On Development of Snake-like Robots to Search a Victim Left inside Debris," IEEE Robotics and Biomimetics (ROBIO 2004). pp.65-69, 22-26 Aug. 2004
- [68] M. Naoki, O. Kazunori, T. Satoshi, K. Hiroyuki, "Development of Small-Size Multi Camera System for Snake-like Robot, and Display of Wide View-Angle Image," IEEE Safety, Security and Rescue Robotics (SSRR 2007), pp.1-6, 27-29 Sept. 2007.
- [69] S. Hirose, E. Fukushima, "Snakes and strings: new robotic components for rescue operations," *SICE 2002. Proceedings of the 41st SICE Annual Conference* , vol.1, no., pp. 338- 343 vol.1, 5-7 Aug. 2002

- [70] U. Sanal, A. Erkmen, I. Erkmen, "Logical sensing and intelligent perception based on rough sets and dynamic decay adjustment learning," *IEEE Robotics and Automation (ICRA '02)*, vol.4, pp. 3386- 3391 vol.4, 2002.
- [71] Engineering Services Inc. :<http://esit.com/index.php>. Nov. 2010.