A Hybrid Procedural/Knowledge-Based Approach to the Animation of Human Hand Grasping

by

Chor Guan Teo

B.Eng.(Electrical), National University of Singapore, 1987

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
Master of Science
in the Department
of
Computing Science

© Chor Guan Teo 1994
SIMON FRASER UNIVERSITY
February 1994

All rights reserved. This work may not be reproduced in whole or in part, by photocopy or other means, without the permission of the author.
APPROVAL

Name: Chor Guan Teo

Degree: Master of Science

Title of thesis: A Hybrid Procedural/Knowledge-Based Approach to the Animation of Human Hand Grasping

Examining Committee: Dr. James Delgrande
Chair

Dr. Tom Calvert
Senior Supervisor

Dr. John Dill
Supervisor

Dr. Tom Shermer
Supervisor

Dr. Christine Mackenzie
Supervisor

Dr. David Fracchia
External Examiner

Date Approved: 2/2/1994

ii
SIMON FRASER UNIVERSITY

PARTIAL COPYRIGHT LICENSE

I hereby grant to Simon Fraser University the right to lend my thesis, project or extended essay (the title of which is shown below) to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users. I further agree that permission for multiple copying of this work for scholarly purposes may be granted by me or the Dean of Graduate Studies. It is understood that copying or publication of this work for financial gain shall not be allowed without my written permission.

Title of Thesis/Project/Extended Essay

A Hybrid Procedural/Knowledge-Based Approach to the Animation of Human Hand Grasping

Author: ________________________________________________________________________

Chor Guan Teo

March 2, 1994
Abstract

Although computer animation of articulated figures has been the focus of extensive research in computer graphics, the study of the animation of hand grasping has been rather limited. However, hand grasping has been extensively studied in the fields of kinesiology and robotics.

Traditionally, researchers in the field of robotics have used analytical methods to solve the problem of hand grasping. In recent years, a knowledge-based approach, which uses information obtained from motor control studies, has increasingly gained acceptance. Studies in kinesiology have shown that humans tend to use a pre-determined set of hand configurations. This makes the search for a suitable grasp posture tractable. Thus, it is possible to use the information obtained from these two fields in the development of an approach to the animation of human hand grasping.

In this thesis, the main objective is to investigate and develop tools to support the modelling and animation of human hand grasping. A hybrid procedural knowledge-based approach is used to construct an animation system, which will serve as the platform for determining the effectiveness of these tools.
Dedication

And whatever you do, whether in word or deed, do it all in the name of the Lord Jesus, giving thanks to God the Father through him. *Colossians 3:17*
I would like to thank my supervisors for their help and guidance throughout the project. Special thanks to Chris Welman for his time and patience, for without him, I would not have been able to appreciate and understand the technical intricacies of the Life Forms libraries and inverse kinematics. I would like to thank the members of the Computer Graphics Research Lab at SFU for the lively environment, especially Dave Fracchia who has helped me prepare my seminar, as well as Philip Peterson and Albert Ho, who have taken the time and effort to proofread this thesis. Thanks are also given to Sing-Bing Kang of Carnegie-Mellon University, Christina Lau and Lucy Teh for their guidance and support whenever I needed it. This work was supported in part by the IRIS Network of Centres of Excellence, the Natural Sciences and Engineering Research Council and the Social Sciences and Humanities Research Council.
## Contents

Abstract ........................................................................................................ iii
Dedication ..................................................................................................... iv
Acknowledgements .................................................................................... v
List of Tables ............................................................................................... viii
List of Figures .............................................................................................. ix

1 Introduction ................................................................................................. 1
   1.1 Procedural Knowledge-Based Approach .................................................. 2
   1.2 Human Hand Anatomy ........................................................................... 3
   1.3 Kinematics of the Human Hand ............................................................... 6
   1.4 Modelling of the Hand ........................................................................... 6
   1.5 Prehension ............................................................................................. 8
   1.6 Organization .......................................................................................... 8

2 Literature Survey ........................................................................................ 9
   2.1 Kinesiology ........................................................................................... 9
      2.1.1 Grasp Classification ....................................................................... 9
      2.1.2 Grasp Execution ........................................................................... 15
      2.1.3 Task Requirements and Object Attributes ................................. 18
   2.2 Robotics ................................................................................................ 21
      2.2.1 Analytical Methods ....................................................................... 22
      2.2.2 Knowledge-Based Methods ......................................................... 25
      2.2.3 Robot Hands ................................................................................ 27
   2.3 Computer Graphics ............................................................................... 31

3 Problem Definition .................................................................................... 33
   3.1 Human Hand and Arm Modelling ......................................................... 33
3.2 Grasp and Trajectory Planning ........................................ 34
3.3 Scope of Work .................................................................. 35
4 Implementation .................................................................... 40
4.1 Knowledge-Base Layer ...................................................... 40
4.2 Application Layer ........................................................... 43
  4.2.1 Human Hand and Arm Modelling ............................... 43
  4.2.2 Grasp and Trajectory Generation ............................... 47
  4.2.3 Grasp Algorithms ..................................................... 50
  4.2.4 Hand Shaping and Alignment Algorithms .................. 53
  4.2.5 Sequence Generation and Playback ............................. 55
4.3 Life Forms Layer ............................................................. 55
5 Results and Discussion ....................................................... 58
  5.1 Results ........................................................................ 58
  5.2 Problems Encountered and Solved ................................. 61
  5.3 Discussion .................................................................... 64
    5.3.1 The Posture Library ................................................ 64
    5.3.2 Animation of the Hand ............................................. 65
6 Conclusion ........................................................................ 66
  6.1 Summary ...................................................................... 66
  6.2 Future Work .................................................................. 67
    6.2.1 Object Types .......................................................... 67
    6.2.2 Manipulation of Objects .......................................... 68
    6.2.3 Knowledge-Base ...................................................... 68
    6.2.4 Posture Generation .................................................. 69
    6.2.5 Incorporation of Inverse Kinematics ......................... 70
    6.2.6 Integration into an Animation System Using Life Forms 70
Appendices
  A Grasp Choice Selection .................................................... 71
  B Hand Posture Library ....................................................... 73
Glossary .............................................................................. 75
Bibliography ........................................................................ 83
List of Tables

1.1 Joint Limits of the Hand (from [14]) ........................................ 4

2.1 Classifications of Grasp Postures (from [34]) ............................... 10
2.2 Examples of Prehensile Tasks (from [34]) ..................................... 20
2.3 Definitions of Analytical Measures for Describing a Grasp (from [20]) .... 24

4.1 Coefficient Table for Segment Length (from [14]) .......................... 44
4.2 Coefficient Table for B-axis of the Ellipsoid (from [14]) .................... 45
4.3 Coefficient Table for C-axis of the Ellipsoid (from [14]) .................... 45
# List of Figures

1.1 Human Hand Anatomy ........................................... 5  
1.2 Kinematic Model of the Hand .................................... 7  

2.1 Schlesinger's Grasp Classification (from [55]) .............. 11  
2.2 Napier's Grasp Classification (from [45]) ....................... 12  
2.3 Iberall's Prehensile Classification Based on Opposition Forces (from [34]) 14  
2.4 Reaching Trajectory Path (from [33]) .......................... 17  
2.5 Parameters Affecting the Grasp Posture (from [34]) ........... 19  
2.6 Issues in Analytical Modelling of Grasping and Manipulation (from [20]) 23  
2.7 Maximization of Analytical Grasp Choice (from [33]) ........ 25  
2.8 Cutkosky's Grasp Taxonomy (from [33]) ......................... 28  
2.9 The Stanford/JPL Robot Hand (from [50]) ..................... 30  
2.10 The Utah/MIT Robot Hand (from [35]) ........................ 30  

3.1 Top and Side Views of the Initial Posture of the Hand and Object (Cylinder) 38  

4.1 Animation System Diagram ...................................... 41  
4.2 Opposition Surfaces for Rectilinear and Cylindrical Objects ........................................... 42  
4.3 Hierarchical Representation of the Skeletal Structure ........ 46  
4.4 Ellipsoidal Description of the Hand Segment (from [14]) .... 48  
4.5 Diagrammatic Representation of the Power Grasp Algorithm .................. 52  
4.6 Alignments for Different Grasps ................................. 54  
4.7 Script File for the Skeleton of the Hand ......................... 56  

5.1 Task and Object Attributes ...................................... 59  
5.2 Main User Interface for the Animation System .................. 59
5.3 User Interfaces for the Object and Arm Parameters .......... 60
5.4 Grasp Postures for a Cylinder .................................. 62
5.5 Grasp Postures for a Cube ...................................... 63
A.1 Rules Used for Grasp Choice ..................................... 72
B.1 Examples of the Hand Posture Library .......................... 74
Chapter 1

Introduction

The animation of articulated human figures has been an active research topic in recent years. Examples include work done by Calvert et. al. (Life Forms [16, 17]) on multiple human figure animation, Badler (Jack [4, 5]) on anthropometrically correct human body motions and Magnenat-Thalmann and Thalmann [43, 44] on human behavioural models.

Within the field of human figure animation, emerging research topics include the animation of human locomotion and hand grasping. Both lend themselves to procedural animation, since the movements are highly structured and repetitive. Although there has been much work on the animation of locomotion [4, 11], there has been very little work on grasping (Magnenat-Thalmann and Thalmann [43] and Rijpkema and Girard [49]). However the area of human grasping has been widely studied by those working in the field of robotics (multi-fingered robot hand grasping) and kinesiology.

The study and implementation of hand grasping is a major research topic in robotics. Originally, robot hands were confined to gripper configurations, but they limited the functionality the robot could achieve. Recently, there has been interest in applying human hand studies obtained from kinesiology to implement a more dextrous hand, thus improving the functionality of the robot.

The objective of this thesis is to provide a framework for the animation of human hand grasping using information obtained from kinesiology and robotics. A hybrid procedural knowledge-based approach is used to build this framework with an existing software package,
CHAPTER 1. INTRODUCTION

Life Forms [16] serving as the platform.

The major implementation issues in the development of the framework are: the design and structure of a three-dimensional model of the hand, the trajectory for the reach of the hand, and the selection of a grasp posture suitable for the intended task and target object.

The approach taken in implementing this framework consists of three stages, and is similar to that suggested by Rijpekma and Girard [49], Stansfield et. al. [53] and Tomovic et. al. [58]. The first stage is the identification of the object to be grasped. This information can be easily obtained by storing the object's attributes in a knowledge-base. In the second stage, a suitable grasp posture is selected based on the object's attributes and the higher level task goals. The animation of the movement is then carried out in the final stage.

1.1 Procedural Knowledge-Based Approach

In creating an animation sequence, an animator uses knowledge in three ways:

- There is explicit knowledge which is built into the keyframes. That is, the animator specifies the keyframes which solely determine the entire animation sequence.

- Another type of knowledge is procedural, where the information is built into the algorithms. However, certain parameters are still determined by the animator.

- The third type of knowledge is declarative. An animation system which utilizes this form of knowledge is supported by a knowledge-based (or expert) system. The knowledge is stored in a knowledge-base and a reasoning engine is used to derive the movement.

In this hybrid procedural knowledge-based system, the knowledge for driving the animation is both procedural and declarative. In a procedural system the essence of a movement is incorporated into an algorithm (or procedure), and a desired movement is specified by a set of parameters. These parameters describe the basics of the movement, thus allowing for different instances of a movement. Therefore, the usability of such a system depends greatly
on the choice of these parameters. For instance, parameters which determine the grasping movements are: the velocity of the arm, the task for the grasping motion (picking up a pen is quite different than picking up a glass), and the target object’s attributes (e.g. its size, shape and weight attributes).

Knowledge incorporated in procedural motions can take on a number of different forms. Explicit information includes kinematic constraints such as angular limits on joint rotation and the inter-dependencies of the inter-segmental joints of a finger [3, 10, 59]. Implicit information can be provided through the equations describing the inverse kinematics of the skeletal frame and those defining the dynamics of the movement. Inverse kinematics is particularly helpful in specifying grasping movements where the required motion is specified by a trajectory for the hand. Based on this information, inverse kinematics is used to calculate the joint angles for the shoulder, elbow and the wrist implicitly.

For a knowledge-based approach, the knowledge is stored in a database, and in this thesis, production rules are used for the reasoning engine. Combining all these sources of knowledge, it is possible to build an animation system which produces motion that looks convincingly real. However, the animator usually has to experiment to get the desired results.

1.2 Human Hand Anatomy

The hand is one of the most complex mechanisms in the human body as it has more than 25 degrees of freedom. As a result, there have been many studies on the anatomical structure of the hand [1, 39, 55, 59]. The skeletal components of the human hand include the wrist bones, the palm bones and digit bones. An anatomical representation of the human hand is given in Figure 1.1.

As shown in this figure, each of the four fingers of the hand has three joints. The most proximal joint (or the joint closest to the palm) is called the meta-carpophalangeal joint (MCP) which has two degrees of freedom\(^1\); an adduction-abduction range of approximately 30 degrees and a flexion-extension range of about 120 degrees. The next two joints in

\(^1\)The degrees of freedom of a joint mentioned in this context refer to the number of axes about which this joint can rotate.
the finger are the interphalangeal joints (the proximal interphalangeal (PIP) and the distal interphalangeal (DIP) joints) that have one degree of freedom each. The range of movement for the DIP is 60 degrees while that for the PIP is 100 degrees.

As the thumb is a more complex mechanism, only a simplified explanation is given here. The thumb’s proximal joint is known as the carpometacarpal joint (CMC). It has two degrees of freedom; an adduction-abduction movement of about 120 degrees and a flexion-extension movement of 45 degrees. The next joint of the thumb is the metacarpal-phalangeal joint (MCP) which has 2 degrees of freedom; a flexion-extension range of about 50 degrees and an abduction-adduction range of approximately 30 degrees. The most distal joint of the thumb is the interphalangeal joint (IP) is a hinge joint of 85 degrees.

Table 1.1 gives the degrees of freedom at each joint and the range of movement possible. The structural limits of these joints are dictated by the anatomy and physiological make-up of the hand. Though the hand has over 25 degrees of freedom, some of these degrees are interdependent. This interdependency arises from the muscular inter-connections of the hand.

<table>
<thead>
<tr>
<th>Hand Element</th>
<th>Joint</th>
<th>Degree of Freedom</th>
<th>Flexion-Extension</th>
<th>Abduction-Adduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finger</td>
<td>DIP</td>
<td>1</td>
<td>60</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>PIP</td>
<td>1</td>
<td>100</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>MCP</td>
<td>2</td>
<td>90</td>
<td>60</td>
</tr>
<tr>
<td>Thumb</td>
<td>IP</td>
<td>1</td>
<td>85</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>MCP</td>
<td>2</td>
<td>50</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>CMC</td>
<td>2</td>
<td>120</td>
<td>45</td>
</tr>
</tbody>
</table>

Table 1.1: Joint Limits of the Hand (from [14])
Figure 1.1: Human Hand Anatomy
1.3 Kinematics of the Human Hand

The human hand can be considered to be an articulated structure in order to provide an analytical description of its movements. An articulated structure is composed of rigid body segments connected by rotational joints. A hierarchical model of this structure can be built by modelling the joints as nodes of a tree. Each joint (or node) has its own local coordinate system or transformation matrix. In this way, the overall transformation at the root of this tree is achieved by traversing its branches and multiplying the matrices at each of the nodes. Constraints on the movement of the body segments are introduced because, for any articulated structure, rigid bodies that are connected together will always be constrained in their movements by their interconnections to other parts in the system.

The root joint of this skeleton is the wrist joint and its child joints are the MCP joints of the four fingers and the CMC joint of the thumb (as shown in Figure 1.2). For each finger, the PIP joint is the child joint of the MCP joint while it is the parent of the DIP joint. For such a hierarchy, the effects of the wrist movements are propagated down to the fingers. Thus the fingers are also rotated if the wrist is being rotated.

1.4 Modelling of the Hand

The next step after defining the kinematic structure of the hand is to define its geometry so that it can be displayed visually. There are three different methods of modelling the geometry of the hand: parametric surface representation, implicit surface representation, and polygonal surface representation.

Traditionally, the parametric surface representation has been used as a means of modelling objects. But in recent years, the implicit surface representation is gaining recognition. Researchers such as Blinn [7], Bloomenthal [8] and Wyvill and Wyvill [63, 64] have looked into ways in which implicit functions can be used to represent blended surfaces. Although an implicit surface takes more time to display than a parametric surface, it has the inherent advantage in that it affords the modeller a finer sense of control over the blending and constraints set in the surface. Recently, Bloomenthal and Wyvill [9] have developed techniques which improve the interactivity of such representations. The third form of representation, the polygonal representation, is the most primitive and the easiest to implement. This form
Figure 1.2: Kinematic Model of the Hand
of representation, however, lacks the ability to represent deformable objects or blended bodies easily. Thus cracks may appear on the surface of the object which is undesirable in surface representations. In this thesis, the polygonal representation is used for the modelling of the hand as it is the easiest to implement.

1.5 Prehension

Prehension is defined to be the act of taking hold of an object with the objective of manipulating or transporting it. On a kinetic level, prehension entails the application of forces during interaction with an object. On the kinematic level, prehension involves the orientation and posture of the hand with an appropriate transportation of the hand and arm to a desired location in space.

Another definition of prehension given by Iberall and Mackenzie is "the application of functionally effective forces by the hand to an object, for a task, given numerous constraints" [34]. The constraints on the hand can be either functional or physical. Functional constraints are those that are set by the nature of the tasks. An example is that of an object that must not be dropped. Physical constraints are those that are based on the properties of the object, such as its shape, size and texture. These constraints include the degrees of freedom and the angular limits of the joints. Thus, the basic functions of prehension are to provide a stable grasp, to impart motion to (or manipulate) the object and to gather sensory information (such as the hardness or the temperature of the object) about it.

1.6 Organization

In the next chapter, a review of relevant literature is given, followed by a description of the scope of the work in Chapter 3. Chapter 4 gives an overview of the implementation issues involved in the animation. The results and discussion are given in Chapter 5 while the conclusion and future work are presented in Chapter 6.
Chapter 2

Literature Survey

The analysis of the human hand and its movement can be complex. This is because there are many aspects to this problem, ranging from the study of the motor functions of the hand to the industrial applications of these results (an example is the robot hand). This chapter is an overview of the research into the human hand that has been carried out in the areas of kinesiology, robotics and computer graphics.

2.1 Kinesiology

There is a wide range of literature in the field of kinesiology on the anatomy, physiology, biomechanics and motor behaviour of the human hand and arm. Two issues are central to the study of human hand motion: grasp classification and the planning motions of the hand and arm. These issues are explored in more detail in the following sections.

2.1.1 Grasp Classification

Over the years, there have been many attempts to classify and categorize the possible hand postures from the perspectives of medical, surgical and industrial applications. Table 2.1 gives a summary of these different classifications [30, 34]. Grasp classifications proposed by Schlesinger [51] and Napier [45] are discussed in this section while the taxonomy proposed by Cutkosky [19] and Cutkosky and Howe [20] will be discussed in Section 2.2.2.

Schlesinger’s taxonomy was developed as a means for classifying human postures from
<table>
<thead>
<tr>
<th>Researchers</th>
<th>Posture Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schlesinger (1919)</td>
<td>cylindrical grasp</td>
</tr>
<tr>
<td></td>
<td>spherical grasp</td>
</tr>
<tr>
<td></td>
<td>palmar prehension</td>
</tr>
<tr>
<td></td>
<td>tip prehension</td>
</tr>
<tr>
<td></td>
<td>hook prehension</td>
</tr>
<tr>
<td></td>
<td>lateral pinch</td>
</tr>
<tr>
<td>McBride (1942)</td>
<td>whole hand grasping</td>
</tr>
<tr>
<td></td>
<td>thumb, finger grasping</td>
</tr>
<tr>
<td></td>
<td>palm, digits grasping</td>
</tr>
<tr>
<td>Griffiths (1943)</td>
<td>cylinder grip</td>
</tr>
<tr>
<td></td>
<td>ball grip</td>
</tr>
<tr>
<td></td>
<td>pincer grip</td>
</tr>
<tr>
<td></td>
<td>pliers grip</td>
</tr>
<tr>
<td></td>
<td>ring grip</td>
</tr>
<tr>
<td>Slocum And Pratt (1946)</td>
<td>grasp</td>
</tr>
<tr>
<td></td>
<td>pinch</td>
</tr>
<tr>
<td></td>
<td>hook</td>
</tr>
<tr>
<td>Napier (1965)</td>
<td>power grasp</td>
</tr>
<tr>
<td></td>
<td>precision grasp</td>
</tr>
<tr>
<td></td>
<td>hook grasp</td>
</tr>
<tr>
<td>Landsmeer (1962)</td>
<td>power grasp</td>
</tr>
<tr>
<td></td>
<td>precision handling</td>
</tr>
<tr>
<td></td>
<td>hook grasp</td>
</tr>
<tr>
<td>Iberall And Lyons (1984)</td>
<td>basic power</td>
</tr>
<tr>
<td></td>
<td>modified power</td>
</tr>
<tr>
<td></td>
<td>basic precision/power</td>
</tr>
<tr>
<td></td>
<td>basic precision</td>
</tr>
<tr>
<td></td>
<td>modified prec/power</td>
</tr>
<tr>
<td></td>
<td>fortified prec/power</td>
</tr>
<tr>
<td>Lyons (1985)</td>
<td>encompass grasp</td>
</tr>
<tr>
<td></td>
<td>precision grasp</td>
</tr>
<tr>
<td></td>
<td>lateral grasp</td>
</tr>
<tr>
<td>Cutosky And Wright (1986)</td>
<td>sm diam heavy wrap</td>
</tr>
<tr>
<td></td>
<td>lrg diam heavy wrap</td>
</tr>
<tr>
<td></td>
<td>medium wrap</td>
</tr>
<tr>
<td></td>
<td>adducted thumb wrap</td>
</tr>
<tr>
<td></td>
<td>light tool wrap</td>
</tr>
<tr>
<td></td>
<td>disk power</td>
</tr>
<tr>
<td></td>
<td>spherical power</td>
</tr>
<tr>
<td></td>
<td>hook</td>
</tr>
<tr>
<td></td>
<td>3 finger precision</td>
</tr>
<tr>
<td></td>
<td>2 finger precision</td>
</tr>
<tr>
<td></td>
<td>disk precision</td>
</tr>
<tr>
<td></td>
<td>spherical precision</td>
</tr>
<tr>
<td></td>
<td>tripod precision</td>
</tr>
<tr>
<td></td>
<td>lateral pinch</td>
</tr>
</tbody>
</table>

Table 2.1: Classifications of Grasp Postures (from [34])
the perspective of designing a prosthetic hand. He had identified a set of six different postures (Figure 2.1) based on the parameters of object shape, hand surface (tip, lateral or palmar) and hand shape. For example, the classification for tool handling (cylindrical prehension) is different from that of handling thick flat objects (palmar prehension) because of the differences in the objects.

One of the most widely referenced and used grasp classifications was proposed by Napier [45]. He was the first researcher to present a dichotomy between power and precision grasps (Figure 2.2):

1. A power grasp is a posture in which an object is held in a clasp by partially flexed fingers and the palm. According to Napier, the power grasp is defined by adduction of the thumb at both the meta-carpophalangeal and carpometacarpal joints while the fingers are flexed, forming a plane of opposition to the palm. He has also defined another variant of this grasp (known as the “coal hammer” power grasp) where the thumb is abducted. The difference between these two types of power grasp lies in the movement of the thumb, where an adducting motion results in a high precision element (Napier's power grasp), while an abducting motion results in a low precision
2. A precision grasp is a posture in which the object is pinched between the flexors of the thumb and the fingers. The thumb is abducted at both the meta-carpophalangeal and carpometacarpal joints, forming one jaw of the clamp. The opposing jaw is formed by part or all of the flexors' surfaces of the fingers.

According to Napier, the factors influencing the posture of the hand are the shape and size of the object and the nature of the task. Although the shape and the size of the object may influence the type of prehension used, it is usually the type of task (or the nature of the intended activity) that ultimately determines the grasp posture.

An alternative method of analyzing grasp classification is to consider that prehension
involves at least two forces being applied in opposition to each other against the surfaces of the object. Arbib et. al. [2, 29] have termed these opposition forces, and have used this concept to classify grasps into three basic types (Figure 2.3):

1. Pad Opposition
   For pad opposition, the force lies along an axis roughly parallel to the palm. This provides greater flexibility for the manipulation of an object at the expense of stability and force.

2. Palm Opposition
   For palm opposition, the force lies along a direction roughly perpendicular to the palm. In contrast to pad opposition, palm opposition sacrifices flexibility in favour of stability.

3. Side Opposition
   For side opposition, the force lies along a direction transverse to the palm. This type of opposition is a compromise between flexibility and stability. Side opposition between the fingers is stronger (and less flexible) if the object is held proximally in the fingers, and weaker if held more distally.

Virtual Finger Concepts
To further quantify and describe this grasp classification, Arbib et. al. [2, 29] introduced the concept of a virtual finger. A virtual finger is an abstract representation used for describing a collection of fingers or hand surfaces applying opposition forces. Real fingers group together into a virtual finger (VF). Two virtual fingers apply opposing forces against each other, providing stability to the object that is being grasped.

For palm opposition (Figure 2.3b), the palm is the first virtual finger (VF1) and the four fingers are the second virtual finger (VF2). In Figure 2.3b, the dark line indicates the direction of the opposition axis between the two VFs. The dashed lines show the orientations
CHAPTER 2. LITERATURE SURVEY

b) Palm Opposition

c) Side Opposition

Figure 2.3: Iberall’s Prehensile Classification Based on Opposition Forces (from [34])
of the VF's. For this type of opposition, the VF1 ends anywhere on the palmar surface while the VF2 ends on the meta-carpropalangeal joint of the fingers in consideration.

For pad opposition (Figure 2.3a), the opposition occurs between the thumb (VF1) and one or more fingers (VF2). In this case, the dashed line for VF1 represents the movement of the thumb. The virtual finger vector, VF2, denotes the movement of the fingers relative to the thumb. Both VF1 and VF2 vary in length and orientation as the thumb and fingers flex and extend respectively. Finally, for side opposition, the opposition occurs between the thumb (VF1) and the side of the index finger (VF2). For this type of opposition, the opposition axis (the dark line in Figure 2.3c) is orthogonal to both VF1 and VF2.

Napier's grasp classification can be transformed into this type of opposition classification. For instance, pad opposition is used in a precision grasp while palm opposition is used in the coal-hammer power grasp. The power grasp, as defined by Napier, involves a combination of palm opposition for power and stability and side opposition for direction.

2.1.2 Grasp Execution

The grasp execution involves perceiving specific object properties, selecting an appropriate grasp posture and moving and orienting the hand so that it moves from the initial location to a required position. This process can be discussed at two different levels: trajectory planning and task planning. Trajectory planning refers to the transportation of the hand to the final location while task planning refers to the achievement of some goal or task. The grasp posture is selected during the task planning stage, while the execution of the grasping motions is done in the trajectory planning stage.

Trajectory Planning

Trajectory planning consists of transporting the arm and hand to a certain spatial location and preshaping the hand in anticipation of the grasp. Jeannerod [38, 36] has termed these two processes as the transportation component and the manipulation component, respectively.

The general trajectory of the arm has been shown experimentally to be curvilinear [47,
CHAPTER 2. LITERATURE SURVEY

58]. The hand is first raised from its resting position and then lowered down to the target object. Experiments have shown that the vertical projection of this transport trajectory is a straight line while the horizontal projection is a curve (shown in Figure 2.4). Given this experimental description, it is possible to define this trajectory given the initial position of the hand position, the position of the object and height of this trajectory. However, the selection of these parameters only defines the trajectory in the absence of obstacles.

Experimental results (Jeannerod [36, 37, 38]) have also shown that as the hand moves along this trajectory, it assumes a shape that is suitable for grasping the object. In addition, the width of the hand opening at the end of the preshaping is usually larger than the dimension of the object. These studies also indicate that the preshaping is accompanied by a selection of preferred grasp modes and orientations. There are different approaches for the determination of the orientation of the hand with respect to the object. The approach by Iberall et. al. [29, 32] uses the concept of opposition forces to determine the grasp type and hand orientation. This is discussed in more detail in Section 2.1.3. Tomovic et. al.'s [58] approach is different in that the geometrical relationship between the hand and object is used to calculate hand orientation.

Task Planning

Task planning refers to planning for achieving a certain goal. This results in the selection of a grasp posture which satisfies the intended task or goal. Some of the parameters in this planning are:

- Object Properties

  Objects possess properties which affect the type of hand posture used in the grasping process. These properties can be physical attributes such as the shape and size of an object, or surface properties such as its texture, temperature and hardness.

  During the planning process, only visually-perceived properties are used in determining the grasp posture. Examples of such properties are an object's physical attributes and its distance and orientation with respect to the hand. Other object properties (such as the surface texture or hardness) are determined through direct interactions with the
CHAPTER 2. LITERATURE SURVEY

3D View of the hand trajectory

Top View

Side View

Figure 2.4: Reaching Trajectory Path (from [33])
object (termed haptic perception [61]). Studies have shown that humans can judge an object sizes quite accurately (i.e. they are able to perceive the size of an object and then set the fingers wide enough to grasp it).

- **Knowledge of Task Mechanics**

  Task mechanics refers to the physical forces and motions imparted on an object. Thus, knowledge of these task mechanics refer to the anticipatory knowledge that humans use in predicting the results of an action. An example is when a person is grasping an object, the location of the centre of mass of the object is estimated so that it does not slip out of the hand.

  Thus the selection of a grasp posture involves satisfying both these requirements. The objective of this task planning is then to find a posture that provides a stable grasp and provides the motions that are required for the object and the given task.

2.1.3 **Task Requirements and Object Attributes**

As shown in Section 2.1.2, both object attributes and task requirements are critical in the determination of a grasp posture. Iberall *et. al.* [29, 30, 31, 32, 33, 34] have quantified the relationship between these two requirements by using the concept of opposition vectors and surfaces. Figure 2.5 shows how the object properties and task requirements are described in terms of these parameters. Specifically:

- It is possible to find pairs of *opposable surfaces* on an object which are roughly parallel to each other. The characteristics of each pair of opposable surfaces are their length, the radius of curvature and the mass. Each pair of opposable surfaces provides stable grasping location for the virtual fingers.

- An approach vector is used to represent the directional vector from the hand (wrist position) to the opposition vector.

- An opposition vector exists between each pair of opposable surfaces. The magnitude of this vector is the width of the object between these two surfaces. Associated with this vector is an orientation variable which measures the angular difference between this vector and the approach vector.
Different task requirements lead to varying forces applied on the object and result in different grasp postures. Contributing to these forces are the perceived weight (mass of the object), torques and inertial forces on the object. The posture used by the hand during the task must be able to overcome these anticipated forces.

The hand also has some physical limits on the postures it can attain since every joint in the hand has a different degree of freedom and angular limit. These DOFs and angular limits constrain the direction of available movements around the opposition vector.

In planning a grasp, the hand will usually move in the direction of the approach vector and the virtual fingers will apply forces around a selected pair of opposable surfaces. The alignment of the hand to this pair of opposable surfaces can be determined by noting the direction of the opposition vector. For example, in a precision grasp, the vector between the two virtual fingers is aligned to be parallel to this opposition vector. A more detailed description of this hand-object alignment using the opposition vector is given in Section 4.2.4.
<table>
<thead>
<tr>
<th>Object Attributes</th>
<th>Task Attributes</th>
<th>GRASP POSTURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surface Length</td>
<td>Object width</td>
<td>Size of Force</td>
</tr>
<tr>
<td>Lift Long Steel Cylinder</td>
<td>4 fingers</td>
<td>medium</td>
</tr>
<tr>
<td>Lift Short Cylinder</td>
<td>3 fingers</td>
<td>medium</td>
</tr>
<tr>
<td>Place Wide Short Steel Cylinder</td>
<td>2 fingers</td>
<td>large</td>
</tr>
<tr>
<td>Lift Large Glass</td>
<td>4 fingers</td>
<td>large</td>
</tr>
<tr>
<td>Place Cylinder</td>
<td>4 fingers</td>
<td>medium</td>
</tr>
<tr>
<td>Lift Small Disk</td>
<td>1 finger</td>
<td>small</td>
</tr>
<tr>
<td>Lift Medium Disk</td>
<td>1 finger</td>
<td>medium</td>
</tr>
<tr>
<td>Pick Small Cube</td>
<td>1 finger</td>
<td>small</td>
</tr>
<tr>
<td>Pick Medium Cube</td>
<td>4 fingers</td>
<td>large</td>
</tr>
</tbody>
</table>

I: Index Finger, M: Middle Finger, R: Ring Finger, L: Little Finger.

Table 2.2: Examples of Prehensile Tasks (from [54])

Examples of grasp posture which satisfy the task requirements and object attributes of a specific task are shown in Table 2.2 (obtained from Iberall and MacKenzie [54]). As shown in this table, the object size is measured in terms of its length and width. In particular, the object length is measured in terms of the number of real fingers that can be placed along the surface length of the object and the width is related to the width of the hand opening. The task requirements have been categorized into force and precision. The precision element of a task refers to the manipulability of the task. From this table, it can be observed that Iberall and MacKenzie do not place as much emphasis on the task to determine the grasp posture as does Napier (Section 2.1.1). Instead, both the object attributes and the task requirements are equally important in the determination of the grasp posture.
For large heavy objects (such as lifting a long steel cylinder), palm opposition is chosen since it provides greater power. As the weight of the object decreases (as in lifting a large glass), the grasp posture switches to pad opposition since the amount of force required is smaller. Also, if the size of the object is reduced, the number of real fingers that are mapped to the second virtual finger is decreased. For example, the number of real fingers for virtual finger two (VF2) in lifting a small disk is only one, while three real fingers are used for lifting a large glass.

Expert systems have been implemented for the selection of the grasp posture that satisfy both the object and task requirements. An example of such an expert system is “Grasp-Exp” developed by Cutkosky [19]. An alternative method is to use artificial neural networks. An example of an artificial neural network is by Iberall [31]. The input to her neural networks are the object and task requirements and the output is the chosen opposition (pad, palm or side opposition).

2.2 Robotics

There has been extensive research on mechanical arms and hands conducted in the field of robotics. This has been motivated by the many applications for generic robot grasping in unstructured hazardous environments, in manufacturing environments, and as devices for the assistance of physically disabled persons.

Robots have wide applications in industry and can be designed specifically for the efficient handling of tools and parts. The end effectors typically found at the end of such robot hands are two-fingered grippers which provide a stable encompassing grasp (as in palm opposition), but lack the flexibility to handle a wide selection of tasks. An alternative is, of course, to equip the robot hand with a variety of different grippers for different tasks. But, this can be time-consuming and expensive. In recent years, the approach has been to design a robot hand which models the dexterity of a human hand. Examples are the Stanford/JPL hand, the Utah/MIT hand and the Belgrade/USC hand. These robot hands are discussed in Section 2.2.3. The issues involved in the design of such robot hands are the kinematics and control of the hand, the generation of stable grasps and the system control schemata.

This review will concentrate on the generation of stable grasps given the nature of tasks
and the geometric and physical properties of the object. There are two main approaches to the generation of robot grasps: analytical methods and knowledge-based methods.

### 2.2.1 Analytical Methods

Analytical methods for selecting grasp postures for the hand have been widely studied and implemented in robotics. These methods model hand grasping using the laws of physics; interactions between the hand and the target object are modelled in terms of the forces and motions between them. The issues involved in choosing a grasp based on these methods are shown in Figure 2.6.

From this figure, it can be seen that the manipulation of the robot hand is complex since the system has to take into consideration the kinematics and dynamics of the hand, contact between the hand and object and redundant degrees of freedom. Redundant degrees of freedom occur in cases where there can be more than one solution for the hand configuration given the object. Because the problem can become increasingly complicated, earlier analyses have made the following assumptions to simplify the problem:

1. The hand and objects are modelled as rigid-bodies and the contacts between them are point contacts. That is, it is assumed that the hand will only grasp objects which are not deformable.

2. The contact points between objects and fingers are ideal (i.e. there are no frictional or viscosity parameters involved in the calculations).

3. There are no redundant degrees of freedom (i.e. there is a single hand configuration for grasping the object).

In recent years, some of these assumptions have been relaxed at the expense of greater complexity ([18, 40]). However, in order to keep the analysis tractable, certain assumptions still apply:

1. The objects to be grasped have simplified representations. They are usually treated as rigid geometric primitives.
CHAPTER 2. LITERATURE SURVEY

Kinematics
force/velocity relations
form, force enclosure
singularity & redundancy
contact kinematics (eg: rolling and sliding)

Dynamics
accelerations
stability
impedance/admittance
actuator & drive-train dynamics

Constitutive Relations
joint and link compliance
fingertip deformations
contact properties
friction conditions
object stiffness

Geometry
object shape
local surface geometry
accessibility

Figure 2.6: Issues in Analytical Modelling of Grasping and Manipulation (from [20])
CHAPTER 2. LITERATURE SURVEY

<table>
<thead>
<tr>
<th>Analytical Measures</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compliance</td>
<td>What is the effective compliance (inverse of stiffness) of the grasped object with respect to the hand? The grasp compliance matrix is a function of grasp configuration, joint servos and structural compliances in the links, joints and fingertips.</td>
</tr>
<tr>
<td>Connectivity</td>
<td>How many degrees of freedom are there between the grasped object and the hand? Formally, how many independent parameters are needed to completely specify the position and orientation of the object with respect to the palm.</td>
</tr>
<tr>
<td>Force closure</td>
<td>Assuming that external forces maintain contact between the fingers and the object, is the object unable to move without slipping when the finger joints are locked?</td>
</tr>
<tr>
<td>Form closure</td>
<td>Can external forces and moments be applied from any direction without moving the object when the fingers are locked?</td>
</tr>
<tr>
<td>Grasp isotropy</td>
<td>Does the grasp configuration permit the finger joints to accurately apply forces and moments to the object?</td>
</tr>
<tr>
<td>Internal forces</td>
<td>What kinds of internal grasp forces can the hand apply to the object? Formally, the internal grasp forces are the homogeneous solutions to the equilibrium equations of the objects.</td>
</tr>
<tr>
<td>Manipulability</td>
<td>While not consistently defined in the literature, a useful definition is: Can the fingers impart arbitrary motions to the object? Thus a manipulable grasp must have force closure and a connectivity of 6.</td>
</tr>
<tr>
<td>Resistance to slipping</td>
<td>How large can the forces and moments on the object be before the fingers start to slip? The resistance to slipping depends on the configuration of the grasp, on the types of contacts and on the friction between the object and the fingertips.</td>
</tr>
<tr>
<td>Stability</td>
<td>Will the grasp return to its initial configuration after being disturbed by an external force or moment? At low speeds, the grasp is stable and the overall stiffness matrix is positive definite. At higher speeds, dynamic stability must be considered.</td>
</tr>
</tbody>
</table>

Table 2.3: Definitions of Analytical Measures for Describing a Grasp (from [20])

2. The friction that exists between the hand and object is idealized as it ignores the material properties of the skin and the presence of dirt or moisture.

Table 2.3 shows the qualitative measures that are used to describe a grasp using analytical methods [20]. Figure 2.7 illustrates the problem of determining a grasp choice using these qualitative measures and constraints. There are three types of constraints:

1. Task constraints (the amount of force or motion to be imparted).
2. Object constraints (the shape and size of the object).
3. Gripper constraints arising from the hand (the size of the robot hand and the maximal openings of the fingers).
Choosing a grasp involves selecting the qualitative measures which will optimize these constraints. An example of an analytical method [46] is to find independent regions of contact which constrain the motion of the object.

### 2.2.2 Knowledge-Based Methods

An alternative approach to the analytical method is to use heuristics in choosing an appropriate grasp. These heuristics are derived from knowledge acquired in kinesiology, psychology and cognitive science. Examples of robot hand implementation using the knowledge-based method have been presented by Stansfield [53], Tomovic et. al. [58] and Iberall et. al. [30, 31, 33]. The latter has been covered in Section 2.1.3.

The approach by Stansfield incorporates both vision and knowledge to drive the grasps of the robot hand. A vision system is used to extract information about the views or aspects of the object. This information is then used as input to a knowledge-based system to generate a suitable set of grasps. This knowledge-based system consists of a set of heuristics which are derived from studies done by psychologists and cognitive scientists in human grasping [36, 45]. These heuristics, which are similar to those laid out by Iberall et. al., help to simplify the number of possible grasp sets. Some examples of these heuristics are:
• Humans tend to use a pre-determined set of hand configurations. This reduces the total number of possible hand configurations to a standard set and makes the grasp selection tractable.

• The target object attributes will influence the type of grasp to be used. However, higher-level knowledge about the nature of the task also affects the selection process.

• Using the concept of virtual fingers [30], some of the fingers can be coupled, reducing the number of degrees of freedom the system has to handle.

For the grasp execution of the robot hand, Stansfield has adopted the human hand grasping model that was proposed by Jeannerod [36, 37, 38] (Section 2.1.2) in which the grasp execution consists of two stages: the transport stage and the manipulation stage.

The heuristic approach of Tomovic et. al. [58] is similar to that of Stansfield. The main difference is that no vision system is used to infer the shape of the object. Instead, they make simplifying assumptions about the objects to be grasped. One of their assumptions is to represent objects of arbitrary shapes by standard geometrical primitives. For example, a glass of water is represented by a cylinder in their system.

**Cutkosky’s Grasp Taxonomy**

Cutkosky [19] developed a grasp taxonomy based on studies of human grasping from the standpoint of manufacturing needs. This approach is different from Stansfield [53, 54] and Tomovic et. al. [58] in that it is not heuristically based. This taxonomy (Figure 2.8) is based on Napier’s (Section 2.1.1) dichotomy between power and precision grasps. This taxonomy shows how task requirements and the object geometry determine the selection of a grasp. The first decision in selecting a grasp from this taxonomy is to decide between a power grasp or a precision grasp. Once this choice is made, a combination of task-related and object geometry factors comes into play. For example, if a power grasp is chosen and the object needs to be clamped (or totally constrained) and is circular in shape, a circular grasp is
CHAPTER 2. LITERATURE SURVEY

selected. As the hierarchy is traversed, the nature of the task and the object geometry play increasingly and equally important roles in the grasp posture selection process.

Looking at the taxonomy, one can see that the grasps become less powerful and the grasped objects smaller as the tree goes from left to right. Thus, “Heavy Wrap” grasps are the most powerful but the least dextrous while the “Prismatic Grasps” are the most dextrous but have the least power. Also as this tree is traversed from the root to the leaf-nodes, the trend is to go from general considerations to more specific identifications of the object geometry and the task. For instance, at the root of the tree, the main consideration is whether the grasp will provide the required restraining forces. At the lowest level (or the leaf-nodes), the choices are based on geometrical descriptions of the object. As mentioned on page 21, an expert system, “Grasp-Exp”, has been developed to use the information in this taxonomy.

2.2.3 Robot Hands

This section presents some robot hands that have been developed in the research area of robotics. The robot hands that are discussed are the Belgrade hand (Bekey et. al. [6]), the Stanford/JPL hand (Salisbury and Craig [50]) and the Utah/MIT hand (Jacobsen et. al. [35]). Additional information about these robot hands can be found in Grupen et. al. [28].

The Belgrade hand is an anthropomorphic robot hand. The hand has five fingers which can be used in two modes: a three-finger mode or a five-finger mode. The goal in designing this hand was to develop a robot hand which was suitable for grasping and capable of autonomous shape adaptation. This robot hand is built based on the paradigm of reflex control. Reflex control is a non-numeric form of control which hypothesizes that control in humans is based on sensory data. Thus, the grasp is executed by using the stored input pattern data to determine what is kinematically natural for the hand. The unique features of the Belgrade hand are:

- The motions of the finger segments are inter-connected. This is different from the Stanford/JPL hand or the Utah/MIT hand where each finger segment is individually controllable.

- The hand is designed for autonomous shape adaptation. This enables the hand to
Figure 2.8: Cutkosky’s Grasp Taxonomy (from [33])
CHAPTER 2. LITERATURE SURVEY

provide firmer grasps.

The advantage of the Belgrade hand is that it is suited for grasping tasks while its main disadvantage is that it is limited to the class of manipulation for which humans employ reflexive control.

The Stanford/JPL robot hand was built based on anthropomorphic considerations and kinematic and control issues. The effectiveness of the manipulator was evaluated based on the number of fingers, the number of links per finger and the types of contact between the finger and the object. The robot hand was designed based on the mobility and the connectivity of the hand-object system. Mobility refers to the number of independent parameters that are required to specify the state of the mechanism. Connectivity refers to the number of parameters required to specify the relative position of the two objects (the hand and the object). The final design of the Stanford/JPL hand is a three-fingered robot hand with three links per finger and contacts with three degrees of freedom.

The Utah/MIT hand was developed in 1982 as an experimental investigation into machine manipulation. The present version has three four degrees-of-freedom fingers and one four degrees-of-freedom thumb. One of the main paradigms of the hand is that it is designed to be a anthropomorphic hand model. The reasons for this approach are:

- It is useful from an experimental viewpoint as it allows the researcher to compare the robot hand’s movement with a human hand’s natural movements.

- There are possible applications in the field of teleoperations.

In order to model the human hand as closely as possible, the Utah/MIT hand implements two reflex motions which are observed in the human hand:

- **Proximal stiffening:** contacts with the environment (for example: the object to be grasped) will cause the proximal joints to stiffen.

- **Distal curling:** if a contact point is established, all the distal joints will curl around this point.

Figures 2.9 and 2.10 show the configurations of the Stanford/JPL hand and the Utah/MIT hand respectively.
CHAPTER 2. LITERATURE SURVEY

Figure 2.9: The Stanford/JPL Robot Hand (from [50])

Figure 2.10: The Utah/MIT Robot Hand (from [35])
CHAPTER 2. LITERATURE SURVEY

2.3 Computer Graphics

There has only been limited work in computer graphics on the animation of human hands. Most of the research in this area has concentrated on the animation of the entire human body [16, 17], on the dynamics or kinematics of walking [4, 11] or on the reconstruction of the hand model from collected data [57].

The basic approaches in the animation of jointed bodies involve either dynamics or kinematics. Dynamics refers to the calculation of motion from forces and torques acting on masses. In treating the rigid bodies as masses, physical laws are used to find out the motions which these bodies undergo. Kinematic modelling of an object refers to the study of its geometry without regard to the external forces acting on it. In this case, the object is defined with respect to a fixed reference coordinate system as a function of time.

There are two ways of handling the kinematics: forward or inverse kinematics. The forward kinematics problem involves finding the geometry of the articulated body given the angles of the joints. The inverse kinematics solution, on the other hand, approaches the problem by trying to find the joint angles of the body given the geometrical position of one or more joints.

Rijpkema and Girard [49] have combined knowledge gained from kinesiology and robotics in designing an animation of the human hand. They have defined a kinematic model of the human hand and arm and have used inverse kinematics to animate it. Their system provides some user-interactivity by allowing an animator to define a hand posture interactively. The motion planning of the arm and hand is done using information from studies done on trajectory planning (see Section 2.1.2).

In their system, the grasping behaviour of the hand can either be defined by the system or by the user. For automatic generation of the grasping behaviour, there are certain characteristics which govern the posture of the hand. These characteristics are obtained from a knowledge-base in the system. Using information from this knowledge-base, the system is able to generate the desired hand posture given the object attributes. Alternatively, the user can define the type of hand gesture to be used in the grasp execution. This allows more flexibility in the animation.
Other researchers who have constructed prototypes of a human hand and animated grasping are Magnenat-Thalmann and Thalmann [27, 43, 44]. The main focus of their research was on the use of finite element analysis and keyframing methods to model and animate the human hand.
Chapter 3

Problem Definition

The main objective of this thesis is to investigate and develop tools to support the modelling and animation of human hand grasping. In this chapter, the steps and problems involved in achieving this goal are presented. The scope of the thesis is then defined by describing the algorithms that are used to implement this animation system. Certain assumptions are also made in order to limit the scope. The major problems that must be solved to achieve this goal are: the modelling of the physical structure of the hand and arm, the generation of the hand and arm movement and the design of a system to implement a reasonable grasp strategy.

3.1 Human Hand and Arm Modelling

The human hand and arm are articulated skeletal structures. To model them properly, we must consider the following:

- The measurements of the skeletal segments.
- The definition of a skeletal structure. This definition requires a description of the joint types between each pair of skeletal segments (or links). For modelling purposes, there are three possible joint types:

  1. Hinge joint - 1 degree of freedom
  2. U-joint - 2 degrees of freedom
  3. Ball-joint - 3 degrees of freedom
CHAPTER 3. PROBLEM DEFINITION

- The attachment of geometric objects to the skeletal structure to generate a solid model of the hand.

3.2 Grasp and Trajectory Planning

As shown in Chapter 2, the generation of grasps is a complex problem since the hand is a dextrous skeletal structure. There are three steps in solving this problem: selection of a suitable grasp posture (grasp planning), movement of the hand and arm to the required location (trajectory planning), and detection of contacts between the hand and the object (hand-object contact).

- Grasp Planning

Grasp planning involves the selection of a suitable grasp posture satisfying the nature of the task and the target object. This selection could be complex as there are many postures a hand can attain for a given task. However, studies have shown that in practice, the hand's posture is chosen from a relatively small set. Thus, the strategy is to select a suitable posture from this reduced set.

There are two ways to approach this grasp selection: analytical (Section 2.2.1) or knowledge-based (Section 2.2.2). The main disadvantage of the analytical method is that the problem can become too computationally intensive if certain simplifying assumptions are removed. Thus, in recent years, the knowledge-based method is increasingly being used as an alternative method. Some of the more well-known approaches are the Grasp-Taxonomy by Cutkosky and Howe [19, 20] (Section 2.2.2) and the Opposition Forces concept by Iberall et. al. [29, 30, 31, 32, 33, 34] (Section 2.1.3). All of these approaches acknowledge that the nature of the task plays an important role in the selection of a grasp posture. Other equally important factors are the object attributes and the spatial position and orientation of the object.

- Trajectory Planning and Movement

The movement of the hand from its initial position to a specific location in space requires the calculation of the trajectory of the hand and arm. Besides this, the
orientation and shaping of the hand need to be determined at the same time.

Studies in the robotics literature [49, 58] show that this trajectory is usually of a predictable shape (a parabolic trajectory). Thus it is reasonable to assume such a trajectory shape for the reach. Besides this trajectory movement, the hand also orients and shapes itself along this path such that it is able to provide a stable grasp at the end of its motion. The type of shaping depends on the posture which is selected for the grasp. The hand approaches this posture as it moves along the trajectory.

- **Hand-Object Contact**
  
  The first two steps of the grasp generation are concerned with the selection of a suitable grasp posture and motions of the arm and hand to a required location. In the final phase, the hand comes into contact with the object so that it holds the object firmly within its grasp.

### 3.3 Scope of Work

The main objective of this thesis is to explore methods for the animation of knowledge-based grasping for the human hand and arm. The first step is the development of an anthropometrically feasible model of the hand and arm. Different hand sizes can be generated by changing the external hand width and length measurements. Having developed the model, the steps for generating an animation are as follows:

1. Receive a command to perform a grasp task.

2. Check the position of the hand and the target object. The system will have this knowledge as well as a physical description (such as size and shape) of the object.

3. Select a suitable grasp. Napier's [45] precision and power grasp classifications are used to describe the grasp postures. To provide a more analytical description, a simplified version of the opposition force concept by Iberall et. al. [29] is used. This method provides an elegant way to map the task and object requirements into a suitable grasp
CHAPTER 3. PROBLEM DEFINITION

posture (as shown in Section 2.1.3). This mapping will be implemented using a small rule-based system.

4. Calculate the shape of the hand needed to grasp the target object. This is done using the above rule-based system. This depends on the intent of the grasp as well as the physical properties of the object. For instance, different grasps will be used depending on the size of the object.

5. Calculate a trajectory for the arm and hand movement. As mentioned, this movement is assumed to be parabolic. The palm of the hand is moved along this trajectory while inverse kinematics is used to find the joint angles of the articulated structure depending on the position and orientation of the hand in space. Inverse kinematics is used as it provides a computationally efficient way to solve the problem of finding these joint angles.

6. Adjust the size of the hand to accurately grasp the object. Depending upon the grasp posture, different algorithms are used to determine the contact areas between the hand and the object. The algorithms that are used are either the precision grasp algorithm or the power grasp algorithm.

7. Terminate the animation when the object is within the grasp of the hand and the hand attains its desired pose.

A polygonal model is used to represent the palm of the hand and the arm. The finger segments are treated as ellipsoids which are also represented as polygons. As mentioned, the grasp postures implemented are the power grasp and the precision grasp. A power grasp is used for tasks such as handling of tools that are usually cylindrical in shape. A precision grasp is used for tasks that require more flexibility and for rectilinear objects. The power grasp algorithm is usually more straight-forward because the animation of a precision grasp involves a more detailed study of the behavioural model of hand animation. This involves studying how different hand configurations are related to the physical attributes of the
target object and the intended task. Studies by Tomovic et. al. [58] and Iberall et. al. [29, 30, 31, 32, 33, 34] describe different grasp strategies for robot hands in a manufacturing environment. This information can be applied to the animation of a human hand.

Certain assumptions are made when designing the flow of this animation system. They are:

- A simple rule-based system is used for determination of the grasp posture.
- The object is within reach of the hand without moving the shoulders.
- The objects that are represented in the system are simple geometric primitives shapes. These are restricted to cylinders and rectilinear solids.
- There are no obstacles in the trajectory path between the initial location of the hand and the target object.
- A list of predefined postures (in the form of a posture library) exist in the animation system. A predefined posture is used for hand shaping motions when executing a precision grasp. These predefined postures are built using the Life Forms package (see Appendix B).
- A knowledge-base on the object attributes is available in the system. Thus, no vision system (as in Stansfield [53, 54]) is required for acquiring information about the object attributes.
- The initial posture of the hand is: the four fingers are fully extended, the thumb is fully extended and abducted, and the palm is faced down on the same surface as the object (Figure 3.1).
- The tips of the fingers, including the thumb, are used as contact points for detecting collision with the object.
- The generated animation sequence is stored as keyframes as the lower layer of this implementation is a keyframed animation system (Life Forms [16, 17]).
- The animation stops after the required hand posture is achieved. This system does not consider the manipulation of the object after it has been grasped by the hand.
Figure 3.1: Top and Side Views of the Initial Posture of the Hand and Object (Cylinder)
In summary, the animation tools to be implemented comprise three stages.

1. The task initialization stage (grasp planning) — where the rule-based system uses the knowledge-base to determine the type of grasp posture.

2. The target approach stage (trajectory planning) — where the trajectory of the hand and arm is generated. The grasp posture from the task initialization stage is used to select a suitable grasp position. A grasp position is the position of the hand such that, for the given posture, the fingers are in contact with the object. The hand is shaped to assume the selected posture at the same time when it is moved along the trajectory.

3. The grasp execution stage (hand-object contact) — where the fingers will enclose the actual object.
Chapter 4

Implementation

This chapter describes the implementation of the approach taken to model the hand and its movements. The implemented animation system can be separated into three different layers (Figure 4.1):

- Knowledge-Base Layer
- Application Layer
- Skeletal (Life Forms) Layer

These layers will be discussed in more detail in the following sections. The main concepts and the approach taken here parallel work that has been done by Rijpkema and Girard [49] and Stansfield [53], where a knowledge-based approach is preferred to analytical methods.

4.1 Knowledge-Base Layer

Studies by Iberall and Mackenzie [34] have shown that the factors affecting the selection of a grasp posture are dependent on the target attributes of the object. For instance, a power grasp will usually be used to pick up a heavy cylindrical object while a precision grasp is preferred for picking up a small object. The nature of the task is also an equally important factor. A person will use a power grasp when lifting a glass to drink from it while he or she will use a precision grasp to move it to another location. The knowledge-base layer contains information about both the object attributes and the nature of the task. The attributes that are stored are the dimensions of the object, its spatial position, orientation and its
CHAPTER 4. IMPLEMENTATION

Task Input: Example: Lift A Cylinder

Knowledge Base Layer

Assumptions:
- Simple Object Primitives
- Objects within reach
- No Obstacles between hand and object
- Prior Information about object is known

Object Attributes:
- Cylinder
- Cube

Object Attributes
- Dimensions
- Location
- Opposition Surfaces

Hand Attributes:
- Finger Segments Measurements

Arm Attributes:
- Height of Reach Trajectory

Goals:
- Input from Task Level

Posture Library:
- Predefined Postures

Rule-Based System
for Hand Posture Selection

Application Layer

Obj-Hand Contact Algorithm
Final Grasp Contact
Build Hand Model
Hand Shaping and Alignment
Sequence Store/Playback

Generation of Hand and Arm Trajectory

Interface to Skeletal Layer

Skeletal Layer

Skeletal Collision Detection Sequences
Splines Inverse Kinematics
-Quaternions

Figure 4.1: Animation System Diagram
(A) Possible Opposition Surfaces for a Rectilinear Object

(B) Possible Opposition Surfaces for a Cylinder (Approximated as a rectilinear object)

Figure 4.2: Opposition Surfaces for Rectilinear and Cylindrical Objects

shape (cylindrical or rectilinear). In addition, the possible pairs of opposition surfaces of the object are stored. Recall that pairs of opposition surfaces are those that provide stable grasps depending on the location, orientation of the object and nature of task. For a rectilinear object, there are three possible pairs of opposable surfaces that are stored in the knowledge-base (the three pairs of parallel faces). There can be infinitely many possible pairs of opposable surfaces for a cylinder. To simplify the problem, a cylinder is treated also as a cube when calculating the number of opposable surfaces, thus reducing the number to three. Figure 4.2 shows the possible pair of opposition surfaces for both the rectilinear and the cylindrical objects.
The knowledge-base also contains information about the attributes of a task (force and precision) as well as attributes of an object. Examples of the information stored are given in the first four columns of Table 2.2 (page 20). A small rule-based system is used to infer the required grasp posture from this information. For a precision grasp posture, the number of fingers for the grasp is also determined. This selection of the number of fingers depends on the surface length of the object compared to the length of the hand as well as the object width. Some of the rules used in the determination of the grasp mode are:

- Maximize the number of fingers used as this helps to provide a more stable grasp.
- Use the dimensions of the object to limit the possible grasp postures.
- Favour the finger closer to the thumb as it provides greater stability (i.e. for a precision grasp which requires two fingers, the thumb and the index finger will be favoured over the other configurations).

Appendix A.1 shows the existing rules in the rule-based system. The selected grasp posture information is then passed to the Application Layer.

4.2 Application Layer

In this layer a three-dimensional model of the human hand is built and animated. The elements in calculating the animation are: the selection of a grasp position, the computation of the arm trajectory and the algorithms for implementing the selected grasp posture.

4.2.1 Human Hand and Arm Modelling

The first step in this layer is the generation of a 3D model of a human hand. The finger segments are represented by ellipsoids and the palm by a polygonal object. The measurements used for this model are obtained from Buchholz and Armstrong [13, 14].

Generation of Lookup Table for Hand Modelling

Buchholz has collected anthropometric data and has correlated this information to the geometrical structure of an ellipsoid for each finger segment. Tables 4.1 to 4.3 show the coefficient relations required to build the hand model using an ellipsoidal representation based on external anthropometric measurements.
As seen in Table 4.1, given the external width and breadth measurements of a human hand, it is possible to generate all the feasible finger segment lengths. This relationship can be expressed mathematically as:

\[ s_{ij} = A_{ij} \times HL \]

where
\( i \) = the segment row in Table 4.1,
\( j \) = the digit column in Table 4.1,
\( s_{ij} \) = the required finger segment length,
\( A_{ij} \) = the correlated coefficient in Table 4.1 and
\( HL \) = the measured hand length.

A lookup table based on these coefficients is created to generate finger segments given any hand measurements.

**Skeletal Model Presentation**

Given the kinematic model of the hand (shown in Figure 1.1) and the constraints which limit each joint, a skeletal model of the hand can be built. This model is built using the
### Chapter 4. Implementation

#### Coefficients (Bij) for Determining Ellipsoidal Semi-axis Breadth of Segments from Hand Breadth

<table>
<thead>
<tr>
<th>Segment</th>
<th>Digit</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distal</td>
<td>I</td>
<td>.113</td>
<td>.093</td>
<td>.094</td>
<td>.091</td>
<td>.084</td>
</tr>
<tr>
<td></td>
<td>II</td>
<td>.108</td>
<td>.103</td>
<td>.100</td>
<td>.093</td>
<td>.087</td>
</tr>
<tr>
<td></td>
<td>III</td>
<td>.150</td>
<td>.104</td>
<td>.101</td>
<td>.097</td>
<td>.092</td>
</tr>
<tr>
<td></td>
<td>IV</td>
<td>---</td>
<td>.250</td>
<td>.250</td>
<td>.250</td>
<td>.250</td>
</tr>
<tr>
<td></td>
<td>V</td>
<td>---</td>
<td>.200</td>
<td>.183</td>
<td>.173</td>
<td>.164</td>
</tr>
</tbody>
</table>

Table 4.2: Coefficient Table for B-axis of the Ellipsoid (from [14])

#### Coefficients (Cij) for Determining Ellipsoidal Semi-axis Depth of Segments from Hand Breadth

<table>
<thead>
<tr>
<th>Segment</th>
<th>Digit</th>
<th>I</th>
<th>II</th>
<th>III</th>
<th>IV</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distal</td>
<td>I</td>
<td>.091</td>
<td>.078</td>
<td>.082</td>
<td>.079</td>
<td>.072</td>
</tr>
<tr>
<td></td>
<td>II</td>
<td>.107</td>
<td>.092</td>
<td>.095</td>
<td>.089</td>
<td>.086</td>
</tr>
<tr>
<td></td>
<td>III</td>
<td>.147</td>
<td>.106</td>
<td>.114</td>
<td>.111</td>
<td>.099</td>
</tr>
<tr>
<td></td>
<td>IV</td>
<td>---</td>
<td>.200</td>
<td>.183</td>
<td>.173</td>
<td>.164</td>
</tr>
<tr>
<td></td>
<td>V</td>
<td>---</td>
<td>.200</td>
<td>.183</td>
<td>.173</td>
<td>.164</td>
</tr>
</tbody>
</table>

Table 4.3: Coefficient Table for C-axis of the Ellipsoid (from [14])
existing skeletal structure definitions for the Life Forms software package [16, 17]. Figure 4.3 illustrates the hierarchical skeletal structure of the hand.

A description of this skeletal structure has been given in Section 1.3. The definition of this skeletal model is given in a script file which defines the hierarchical structure of the hand as well as the constraints of each joint and its initial orientation. A description of the implementation of this script file is given in Section 4.3. There can be variable parameters associated with this skeletal description. An example of such a parameter is the joint offset of each of the joints in the hand. This offset corresponds to the segment lengths of the fingers and varies depending on the hand length and breadth that is being input to the system.

The information given in the skeletal model and the lookup table can then be combined together to generate an ellipsoidal model of the hand. Figure 4.4 shows an ellipsoidal description of a hand segment. The parameters for describing the ellipsoidal geometry are: minor and major axes ($b$ and $a$ axes) and thickness ($c$ axis). These parameters are found
using the coefficients in Tables 4.1 to 4.3. The mathematical relationships are as follow:

\[ a_{ij} = 1.1/2.0 \times sl_{ij}, \]

\[ b_{ij} = B_{ij} \times HB, \]

\[ c_{ij} = C_{ij} \times HB. \]

and the equation of the ellipsoid is

\[ \frac{(x - r/2)^2}{a^2} + \frac{(y + l)^2}{b^2} + \frac{z^2}{c^2} = 1 \]

where
\[ i = \text{the segment row in Table 4.2 and Table 4.3}, \]
\[ j = \text{the digit column in Table 4.2 and Table 4.3}, \]
\[ sl_{ij} = \text{the required finger segment length (as calculated on page 44)}, \]
\[ a_{ij} = \text{the required ellipsoid major axis for finger segment } ij, \]
\[ b_{ij} = \text{the required ellipsoid minor axis for finger segment } ij, \]
\[ c_{ij} = \text{the required ellipsoid thickness for finger segment } ij, \]
\[ B_{ij} = \text{the correlated coefficient in Table 4.2}, \]
\[ C_{ij} = \text{the correlated coefficient in Table 4.3}, \]
\[ l = \text{displacement of the segment length from the major axis of the ellipsoid, and} \]
\[ HB = \text{measured hand breadth.} \]

Only the fingers of the hand are modelled in ellipsoidal geometry; the palm of the hand is modelled as a polyhedron. The ellipsoids for the finger segments are built on-the-fly when the system is started up while a standard palm polygonal definition was built using the Vertigo modelling software [60].

### 4.2.2 Grasp and Trajectory Generation

Once a model of the human hand is built, the next phase is to search for a best grasp position and to generate the arm trajectory. The search for a grasp position is related to finding a suitable pair of opposition surfaces on the object for the required grasp posture.
This search for a given target object is enormous. In this case, it will be time-consuming to find a correct grasp position. One of the reasons for this is that the hand has many degrees of freedom. For example, the hand that is built in this system has a total of 24 angular DOFs. In particular, each finger segment has 4 angular DOFs, the thumb has 5 angular DOFs and the wrist has 3 angular DOFs. As mentioned in Section 2.2.2, it is possible to limit this large search space to a small set of grasp postures using heuristics. A search is made through this space to select an optimized pair of opposition surfaces for grasping. The requirements for selecting a suitable grasp position are the availability of a pair of opposition surfaces and an optimal trajectory path for this position.

- **Availability of the pair of Opposition Surfaces**

  The conditions for testing this are:

  1. The contact surfaces must be reachable by the arm.

     The method to test this is to calculate the distance from the shoulder to each of the surfaces. If for any pair of opposition surfaces, the distance is greater than the arm length, this pair is discarded as a possible solution.

  2. The spread of the hand compared to the target object size.
The spread of the hand is a measure of the distance between the tip of the thumb to the tip of any other finger. The distance between the opposition surfaces must not be greater than the maximum hand spread.

3. The higher level task goals.
   Some of the opposition surfaces may be preferred due to the nature of the task. The task attributes (precision and force) are contained in the knowledge-base layer.

- Optimal Trajectory Path

The selection of the best grasp position must also take into account the motion of the arm. For example, the optimal grasp may be one which is reached by the minimum energy path. Some researchers have used numerical methods [62] or dynamic programming [25] to calculate this. However, these methods are costly in terms of computation and complexity. An alternative method is to use a heuristic approach which has been used in the implementation of the animation system. This approach tries to minimize the sum of the translational and the rotational components between the initial and final hand positions. The translational component is the distance between the initial and final wrist position. The rotational component is calculated by using the concepts of quaternion formulation [48, 52].

Once the grasp position is selected, the next step is to generate the trajectory of the arm. As shown in Section 2.1.2, this trajectory has a predictable shape. The chosen default shape is that of a parabolic curve. This default trajectory is defined such that the trajectory's movement is parabolic in the vertical plane and is linear in the horizontal direction given the initial conditions of the hand as described in Section 3.3. This trajectory is converted to a 3D spline using routines from the Skeletal Layer. It is then stored as keyframes at constant time intervals in the animation system. The parameters for defining this curve are: the initial position, the final position and the height of the parabolic curve. Inverse kinematics is used to calculate the joint angles of the arm and hand as the hand moves along this trajectory. To provide more flexibility for the animator, this trajectory may be changed interactively.
4.2.3 Grasp Algorithms

Depending upon the types of grasps involved, different algorithms are implemented for performing the grasp execution. There are basically two types of algorithms: the power grasp algorithm and the precision grasp algorithm.

Power Grasp Algorithm

Two different power grasp algorithms (due to Buchholz [14]) have been implemented for different thumb postures. The difference between these two algorithms is in the movement of the thumb.

The first power grasp algorithm abducts the thumb for increased power (Napier’s coal hammer power grasp) while the second algorithm adducts the thumb for increased precision (Napier’s power grasp). In the first algorithm, the thumb segments are wrapped around the object while in the second algorithm, the thumb MCP and IP joints are fully extended and the CMC joint of the thumb is flexed until the distal phalangeal segment is in contact with the object.

The next step after determining the thumb joint angles is to determine the finger-flexion angles. This step is the same for both power grasp algorithms. The algorithm for the fingers begins by flexing the proximal joints and proceeds by “wrapping” the hand around the object. Flexion is terminated when the distal segment contacts the surface of the object.

Figure 4.5 shows how a finger wraps around a cylinder. The cylinder is positioned such that it lies alongside the palm of the hand. The algorithm starts by first flexing the MCP joint till the proximal phalange is in contact with the cylinder surface. Contact tests are also carried out for the middle and distal phalanges as there is the possibility that they will contact the object surface before the proximal segment. If this happens, proximal stiffening is carried out (similar to that of the Belgrade hand shown on page 29) where the proximal joints will stiffen. Otherwise, the same steps are repeated for the PIP and DIP joints.

Studies have shown that there are some interdependencies between the segments of a finger [15, 36]. It has been shown that it is almost impossible to move the DIP without moving the PIP and vice-versa. Empirical studies [41] have determined that there is an
almost linear relationship between these two joints given by:

\[ DIP_j = 2/3 \times PIP_j \]

where

- \( DIP_j \) is the amount of angular rotation for the DIP joint of the jth finger, and
- \( PIP_j \) is the amount of angular rotation for the PIP joint of the jth finger.

This relationship is incorporated into the wrapping routine by moving the DIP by the specified fraction of an angle when the PIP joint is flexed.

**Precision Grasp Algorithm**

The precision grasp algorithm is implemented using the concepts of opposition surfaces and virtual fingers [2, 30, 32]. The input to this algorithm is the selected pair of opposition surfaces (as calculated in Section 4.2.2) and the number of real fingers mapped to a virtual finger. The information about the real fingers to virtual finger mapping comes from the knowledge-base.

Each pair of opposition surfaces has an opposition vector associated with it whose magnitude is the distance between the two surfaces. The configuration of a precision grasp is such that virtual finger 2 (VF2) opposes VF1 (which is usually the thumb) (Figure 4.6). Any number of real fingers can be associated with VF2.

The vector between the virtual fingers VF1 and VF2 is termed the *pinch line*. There are two parts to this precision grasp algorithm. The first part is to find the magnitude of the pinch line which corresponds to the magnitude of the opposition vector. The second part is to align this pinch line to the opposition vector using quaternion formulation methods. The first part of the precision grasp algorithm is more involved and the steps required are as follows:

1. The required predefined posture for the given grasp is loaded into the animation system. As mentioned in Section 3.3, the animation system contains a list of predefined postures in the posture library (see Appendix B).
CHAPTER 4. IMPLEMENTATION

Initial Posture

Proximal segment wraps around the cylindrical object

The rest of the finger segments follow by wrapping around the object

Figure 4.5: Diagrammatic Representation of the Power Grasp Algorithm
2. An array of pinch line magnitudes is generated, ranging from the initial hand posture magnitude to the predefined posture pinch line magnitude. The desired pinch line magnitude which corresponds to the magnitude of the opposition vector is found by a binary search through this array.

Another consideration in the selection is the depth of the hand. This depth refers to the distance between the palm of the hand and the mid-point of the pinch line. The pinch line for the fingers must be selected such that the distance from its mid-point to the palm must be smaller than the depth of the hand.

**4.2.4 Hand Shaping and Alignment Algorithms**

Studies in kinesiology have shown that the hand preshapes into the desired posture during the reaching motion [36, 38]. Preshaping means that the fingers involved in the grasping motion are first extended to their maximal angles before being flexed to enclose the object. In this animation system, given that the fingers are fully extended in the initial posture (see Section 3.3, only the flexing motion is implemented. This flexing motion is termed *hand shaping* to distinguish it from the actual definition of preshaping.

For the precision grasp, hand shaping can be achieved using the precision grasp algorithm described above. Once the desired grasp posture is defined, this shaping is done by moving the grasp fingers from their initial position to their desired pose (using forward kinematics), until collision takes place between these fingers and the object. The pinch line is aligned to be parallel with the opposition vector.

Similarly, hand shaping for the power grasp is done by flexing all the fingers by system defined angles (0.1 radians) as the hand moves from its initial position to the final position. As the hand moves towards the target object, it orients itself such that the object is encompassed by the fingers at the end of the trajectory. The surface normal vector of the palm is aligned with the selected opposition vector. The vector alignments are implemented using quaternions. Figure 4.6 shows the alignments required for each type of grasp.
CHAPTER 4. IMPLEMENTATION

Power Grasp Alignment

Surface Normal of the palm

Surface Normal of Chosen Opposition Surface

Top View of Object And Hand

Precision Grasp Alignment

Pinch Line

Opposition Vector of the Object

Side View of Object And Hand

Figure 4.6: Alignments for Different Grasps
CHAPTER 4. IMPLEMENTATION

4.2.5 Sequence Generation and Playback

The Application Layer also provides routines for storing and playing back the animation sequences in real-time. The procedurally generated grasping motions are stored in the system as keyframes. This enables the animator to examine the animation sequences. These routines provide a higher and more interactive level of control for the animator than is provided with the sequence library routines in Life Forms.

4.3 Life Forms Layer

The last layer is the Life Forms lower level routines. Life Forms is a package that provides interactive human animation [16, 17]. Its lower levels are used as a platform for this project.

The Life Forms libraries used are:

- **The Parse Library**
  
  The skeletal description of the hand is defined in a script file. This library\(^1\) provides a set of routines to read and write this script file. The script file can be altered within the animation system, as in the case when the hand is built using external anthropometric measurements.

- **The Skeletal and Object Libraries**

  The skeletal library provides routines to build the skeletal structure of the hand given the description in the script file. This set of routines provide means of defining, displaying and animating the hand as it executes the grasping motions.

  Figure 4.7 shows the skeletal description of the hand. From this figure, it can be observed that the joints in this hand skeleton are either 1 DOF joints (hinge joints) or have multiple degrees of freedom (ball-and-socket) which can also be modelled as a series of hinge joints sharing the same location. Associated with each joint are attributes such as: the joint type, the local axis of freedom and the local limits on its range of movement.

\(^1\)Courtesy of Philip Peterson.
# Skeleton definition

define skeleton "hand"
    joint "wrist"
        group "cmc"
            joint "cmc_y"
            joint "cmc_z"
        group "thumb_mcp"
            joint "thumb_mcp_x"
            joint "thumb_mcp_z"
        group "g_thumb_ip"
            joint "thumb_ip"
            joint "thumb_tip"
        end group
    end group
    end group
end skeleton

# Example of a joint definition

define joint "2nd_mcp_x"
    offset -29.12 -92.684 0
    hinge x 0 90
    mass 1
    orientation 5 0 0
    mirror "2nd_mcp_x"
    twistaxis -y
end joint

Figure 4.7: Script File for the Skeleton of the Hand
CHAPTER 4. IMPLEMENTATION

The internal representation of the skeleton comprises an internal state vector which consists of all the joint variable values as well as a translational vector for the whole skeleton. This information is sufficient to completely define the location and orientation of the skeleton in a single frame. The skeleton can then be animated by repeatedly loading the state vector and instructing the skeleton to display itself.

The object library provides functions to attach different types of geometric primitives to the skeletal structure. It also provides routines to manipulate these geometric primitives. Together, the skeletal and object libraries provide the lowest layer of this project.

- **The Sequence Library**

  The sequence library provides functions for saving and retrieving sequences generated by the animation program. A sequence is an abstract data type for storage of the animation data. That is, it stores the animated movement of a skeleton over time.

  A sequence usually consists of a series of keyframes, where each keyframe specifies the state vector of the skeleton at a particular time frame. This library enables the viewer to query the sequence at any frame of the animation. If the sequence frame that is queried is inbetween the keyframes, an interpolated state (between the two keyframes) is generated on-the-fly and returned. The ability to save the sequences using the sequence library also provides the flexibility for the saved sequences to be displayed in the Life Forms package.

- **Collision Detection Routines**

  This collection of routines\(^2\) detects collisions between different objects using binary space partition trees [56]. This set of routines is critical to the animation system, as it is enables the contact between the object and the hand to be detected when the object is being grasped.

\(^2\)Courtesy of Chris Welman.
Chapter 5

Results and Discussion

5.1 Results

The overall animation system consists of three layers (as shown in Figure 4.1). These layers provide an integrated framework for generating different types of grasp postures depending on the type of object and the higher level task goals.

The knowledge-base layer is implemented using a rule-based system where the user selects the type of task goal. Each goal contains the information about the nature of the task as well as generic descriptions of the object attributes. For example, a task goal of “Lift Long Steel Cylinder” will have the task and object attributes set as shown in Figure 5.1. These attributes result in a palm opposition grasp posture (or power grasp).

The knowledge from this layer is then used to drive the lower level animation system for animating the hand posture. There are also some interactive parameters provided for user interactions. Some of these parameters are: the hand trajectory’s parameters (examples are the duration of the animation sequence and the height for the trajectory), a spline trajectory which allows the animator to alter the keyframe positions in 3D space, the attributes of an object (such as the dimensions and position in 3D space) and sequence playback parameters (such as saving, playing back and stretching the sequence files). The user interfaces for these parameters are shown in Figures 5.2 and 5.3.

This application layer is also the level where the parameters for driving the animation
CHAPTER 5. RESULTS AND DISCUSSION

Figure 5.1: Task and Object Attributes

Figure 5.2: Main User Interface for the Animation System
Figure 5.3: User Interfaces for the Object and Arm Parameters
are computed. As mentioned in Section 4.2, this layer calculates the grasp position based on information from the knowledge-base and executes the grasping motions. Forward kinematics is used to drive the hand shaping animation and binary space partition trees [56] are used to detect the object collisions.

As mentioned in Chapter 4, the entire application layer is built on top of the Life Forms layer. It is thus possible to use the Life Forms interface to playback the stored animation sequences. Figures 5.4 and 5.5 show some examples of different grasp postures generated by the animation system based on the input task and object attributes.

5.2 Problems Encountered and Solved

The main problems encountered in the implementation of this framework for animating the hand grasping motions were:

- **Systems Level Integration**

  The overall system consists of three different layers, of which the first two were implemented by the author and the lowest layer using routines from the Life Forms libraries. Thus, a certain period of time was required to understand the functions and structure of the Life Forms libraries in order to smoothly integrate the first two layers.

- **Grasp Strategy**

  One of the main issues in the animation program is the implementation of a strategy for selecting a reasonable grasp posture. Many approaches have been implemented by different researchers over the years, including analytical, knowledge-based, and defining a certain grasp taxonomy based on observations. It was decided to use the knowledge-based method founded on the concept of opposition forces [33, 34] since it provides an elegant yet relatively computationally inexpensive approach. Several earlier methods that were considered were the "Grasp-Exp" by Cutkosky [19, 20] (a taxonomic approach), and the knowledge-based approach presented by Tomovic et. al. [58]. Cutkosky’s approach is more descriptive than analytical, and thus using his taxonomy is not a good idea for synthesizing grasps. Tomovic’s approach is similar
Figure 5.4: Grasp Postures for a Cylinder
CHAPTER 5. RESULTS AND DISCUSSION

Figure 5.5: Grasp Postures for a Cube
to opposition forces. However, his approach ignores the input from higher level task
goals and the grasp posture selection is based solely on object dimensions.

• **Hand Alignment with Object**

The main concern in the animation of the hand grasping is that the animation looks
natural and that the object is grasped within the fingers. This raises the issue of
orienting the hand such that when the final grasp execution takes place, the motion
looks natural and the object can be encompassed by the hand.

This problem has been explored quantitatively in kinesiology, where the alignment
requirements are described by means of the opposition vectors. An alternative ap-
proach which uses a geometric relationship between the hand and the object has been
presented by Tomovic et. al. [58]. It was decided to use the concept of opposition
vectors as a means of hand-object alignment, the required orientations and alignments
are represented in terms of quaternions.

### 5.3 Discussion

Some of the issues involved in the implementation of this system are:

- The posture library.
- Means of animating the hand.

#### 5.3.1 The Posture Library

The posture library (illustrated in Appendix B) contains examples of pre-defined postures
that were created using the Life Forms package. This information is used in the preci-
sion grasp algorithm to determine the required pinch line for grasping the object (see Sec-
tion 4.2.3). Forward kinematics is then used to animate the hand from its initial position
until the final desired pose is achieved.

A major question is whether it is necessary to have a predefined posture library or
whether the precision grasp can be defined in a more fundamental way. There are different
ways of addressing this question. One method is to collect empirical data to statistically determine the rotation joint angles for the standard hand configurations. This method is not considered here as it is not within the scope of this thesis.

Another method is to use a physically-based approach. This physically-based approach obtains information on how to animate the hand from the study of its muscular structure. The disadvantage of using such an approach comes from the complexity of muscular interconnections of the hand. These interconnections result in high computational complexity in the precision grasp algorithm. Presently an approximate kinematic model is used as it works sufficiently well to generate real-time animations.

5.3.2 Animation of the Hand

The ideal solution to the problem of animating the hand and the arm is to use inverse kinematics. However, the inverse kinematics algorithm that was resident in the Life Forms package did not provide a sufficient tool. The animation of the hand requires the satisfaction of the position and the orientation parameters at various points in time. However, the resident inverse kinematics algorithm can only satisfy the position requirements. This results in awkward-looking animations since the hand orientation is incorrect.

Changes have been made to the resident inverse kinematics (with the help of Chris Welman) so that it now satisfies both the position and orientation requirements. Presently this inverse kinematics algorithm is used to drive the movement of the arm while a forward kinematics algorithm is used to perform the shaping of hand.
Chapter 6

Conclusion

6.1 Summary

A framework for the generation of grasps that depends on the nature of the task and the object attributes has been implemented using a hybrid procedural knowledge-based approach. The end result is an interactive animation system that demonstrates the reaching and grasping movements of the hand. The architecture of this animation system is such that it can be integrated into the Life Forms package.

As a by-product, a spline editor has been implemented to allow the user to alter the trajectory interactively. Other options available to the user are:

- Selectable object attributes (dimensions and shapes) and position in 3D space.
- Interactive loading and playback of the animation sequences.
- User-defined duration of animation sequences.

One of the limitations of this animation system is that the knowledge-base is implemented using a small rule-based system. Another limitation is that the objects that are represented in the system are simple geometrical primitives. The animation system also does not consider any further movements of the hand after it has made contact with the object.
CHAPTER 6. CONCLUSION

This thesis is similar to the work done by Rijpkema and Girard [49]. However, the algorithms that are implemented in this animation system differ from their work in the following ways:

- **Power Grasp Implementation**
  Two different power grasp algorithms have been implemented. Girard and Rijpkema have mentioned the power grasp as one of the grasp modes that the user can choose. However, there is no further mention about how the power grasp is being implemented. In the animation system that is implemented in this thesis, the implementation of the power grasp algorithm is different from the precision grasp algorithm.

- **Grasp Choice Selection**
  An attempt is made to fully utilize the knowledge-base stored in the system in order to simplify the selection of the grasp choice. For instance, information about the possible pairs of opposable surfaces of the object is stored as an attribute in the knowledge-base. This reduces the time required to search for the grasp position of the hand.

- **Modelling of the Hand**
  A procedural method for modelling the hand is provided. The measurements of the segments of the hand can be calculated from its external anthropometric measurements.

### 6.2 Future Work

The animation framework implemented in this project is a simple demonstration of what can be done with a knowledge-based procedural approach to human hand grasping. There is definite potential for further work in this area. Some of the potential areas of research include:

#### 6.2.1 Object Types

The present system described in this thesis only allows for grasping of simple geometric objects. This has also been a limitation for robotic grasping using vision based recognition
techniques (as implemented by Stansfield [53, 54]). The system would be enhanced if more geometric shapes (such as conical or spherical shapes) were added to the list of available primitives. Another enhancement would be the addition of asymmetric objects in this database. Examples of asymmetric shapes include objects that have handles (such as a mug) or objects that do not have symmetrical features. However, the amount of complexity required for the grasping of such irregular objects will definitely be greater than the present system.

6.2.2 Manipulation of Objects

This thesis provides the framework for the animation of the hand grasping the object. It does not take into consideration what happens after the object is firmly grasped. Depending on the task, there can be different types of motions after the object is grasped. For instance, after a person picks up a pencil, he or she can use it to write or twirl it. The ways in which a pencil is grasped for these two motions are quite different.

Studies have shown that the manipulation intent of the task also affects the selection of the grasp posture. Indeed, after the object has been grasped, depending on the nature of the manipulation, the grasp posture may be changed to accommodate the manipulation task. Sensory information can be used at this stage to adjust and maintain the grasp.

The inclusion of the manipulability of the object after contact with the object requires extensions to the present algorithms and knowledge-base. The knowledge-base needs to be extended to include intrinsic information about the object. Examples of such information are: the texture, the hardness or the temperature of the object. The algorithm would also need to be altered to include feedback from both the task and object attributes to enable dynamic changes in the grasp posture.

6.2.3 Knowledge-Base

The "intelligence" of the system can be greatly increased by improving the knowledge-base of the system. The knowledge-base can be improved in the following ways:

1. The use of an expert system that is able to take incomplete information about the nature of the task and infer the optimal grasp. At present, a simple rule-based system
is implemented to handle the knowledge-base and is insufficient for handling more complicated cases.

2. A more extensive knowledge-base could handle objects which are irregular geometrical primitives. More complicated tasks can also be handled with a more sophisticated database. The ability to search this knowledge-base and use it competently depends on the implementation of a suitable expert system.

6.2.4 Posture Generation

The present system automatically generates the posture for grasping the object. The information for this grasp posture comes from a pre-defined grasp posture library. The interactivity of the system can be increased by allowing users to define their own postures using the system. That is, the user is able to interactively “drag” the fingers into postures and store them in a posture library. Various controls could be provided:

- Single Finger Control
  
  Each finger can be dragged individually to create the desired posture. The movement can be implemented using inverse kinematics controls which take the inter-segmental dependencies into consideration.

- Virtual Finger Control
  
  One or more of the fingers can be identified as belonging to the same virtual finger group. For instance, in a power grasp, all of the fingers are identified as belonging to the same virtual finger VF2. The user can then drag all the fingers as a single unit. The operations that can be performed on a VF are similar to those for single finger control.

- Hand Control
  
  The highest level of control is that of hand control. Using the functions mentioned above, the user has the ability to build up the required postures to be stored in a posture library. Controls are also given for adding and deleting postures from this
library. Thus, once a posture is added to the library, it can be retrieved for precision grasp shaping (as in Section 4.2.4).

6.2.5 Incorporation of Inverse Kinematics

In this animation system, forward kinematics has been used to drive the movements of the hand while inverse kinematics has been used to drive the motions of the arm. However, one inverse kinematics algorithm that provides functional constraints would allow the system to perform the animation more adequately. Therefore, an inverse kinematics algorithm can be used to drive the reaching motions of the arm while another inverse kinematics algorithm can be used to drive the motions of the hand.

As mentioned in Section 4.2.3, functional constraints on the fingers come into play in this instance as there are inter-segmental dependencies within a finger and inter-digital dependencies within the hand. Use of appropriate inverse kinematics algorithms can reduce the complexity of the system.

6.2.6 Integration into an Animation System Using Life Forms

Since the lowest layer of this animation is based on the Life Forms routines, it is possible to port the animation sequences that are generated by this system to the Life Forms package. This greatly enhances the user interactivity of the system as the Life Forms package provides a better user interfaces than the present animation system. For instance, the generated animation sequence could become part of the grasping motions of a human figure. In this manner, this animation system can be considered to be a tool of the Life Forms package. This is because the grasping movements generated by this system can be ported over to Life Forms where they can be altered by an animator.
Appendix A

Grasp Choice Selection

Figure A.1 shows some of the rules that are used in the selection of the grasp type.
APPENDIX A. GRASP CHOICE SELECTION

IF (object_primitive == CYLINDER) &&
    (precision_type == LARGE)
THEN
    select PAD Opposition
    Decide on VF2 Mapping:
        IF (object_dimensions < 1/4 palm_width)
            VF2 = 1 finger
        ELSE IF (1/4 palm_width < object_dimensions < 2/4 palm_width)
            VF2 = 2 fingers
        ELSE IF (2/4 palm_width < object_dimensions < 3/4 palm_width)
            VF2 = 3 fingers
        ELSE
            VF2 = 4 fingers

IF (object_primitive == CYLINDER) &&
    (precision_type == SMALL) &&
    (object_dimensions == LARGE)
THEN
    select PALM Opposition

IF (object_primitive == CUBE)
THEN
    select PAD Opposition
    Decide on VF2 Mapping

Figure A.1: Rules Used for Grasp Choice
Appendix B

Hand Posture Library

Figure B.1 shows some examples of the predefined postures that are stored in the Posture Library. These predefined postures are generated using the Life Forms system and are loaded into the animation when required.
Figure B.1: Examples of the Hand Posture Library
Glossary

Abduction The movement away from the body.

Adduction The movement of a limb towards the body.

Anthropometry The study of human body measurements especially on a comparative basis.

Anthropomorphism The ascription of human shapes or qualities to non human creatures or inanimate objects.

Articulated Body A body is made up of segments or links (usually rigid) connected by joints. The motion of the segments relative to each other is maybe restricted. A human body can be expressed as an articulated body.

Degrees of Freedom (DOFs) DOFs are the number of independent parameters required to completely specify the positions of every part of the system. For example, a segment or rigid body has 6 DOFs (3 translational coordinates and 3 rotational ones for orientation).

Distal Employed with reference to limbs only, this term refers to a structure being further away from the median plane or root of the limb than another structure in that limb.

Extension The act of bringing the distal portion of the joint into continuity (though only parallel) with the long axis of the proximal portion.

Extensor A muscle the contraction of which tends to straighten a limb.

Flexion The act of bending where there is a decrease in angle between two bones.

Flexor A muscle whose action is to flex a joint.
Kinematics The analysis or description of a body or parts of a body in time and space independent of the forces that cause these movements. It involves the calculation of linear and angular displacements, velocities and accelerations.

Median The midline plane which divides the body into left/right halves.

Medial This refers to a structure being further away from the median plane than another structure in the body.

Proximal Employed with reference to limbs only, this term refers to a structure being closer to the median plane or root of the limb than another structure in the limb origin.
Bibliography


