

Ben-Gurion University of the Negev
The Faculty of Engineering Sciences
The Department of Industrial Engineering and Management

Examination of Constrained Motion on a Hemispherical Surface

Thesis submitted in partial fulfillment of the requirements for the M.Sc. degree

By: Danny Eizicovits

Advisor: Dr. Berman

December 2010

Examination of Constrained Motion on a Hemispherical Surface

Thesis submitted in partial fulfillment of the requirements for the M.Sc. degree

By: Eizicovits Danny

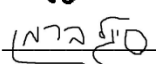
Advisor: Dr. Sigal Berman

Ben-Gurion University of the Negev, Faculty of Engineering Sciences
Department of Industrial Engineering and Management

Author: Danny Eizicovits

Advisor: Dr. Sigal Berman

Chairman of graduate students committee: Prof. Joseph Kreimer


December 2010

Acknowledgments

I wish to express my deepest gratitude to my advisor, Dr. Sigal Berman, which was very helpful and without her wisdom, guidance, support, long discussions and patience this thesis would never have existed.

I would like to gratefully acknowledge my two consultants: Dr. Dario G. Liebermann and Dr. Joe McIntyre, which helped a lot in the building and recording of the movements in the virtual and the physical environment. Also I would like to thank them for their insight full remarks and accurate discussions during the process of analyzing the results. One more special thanks goes to Joe McIntyre who offered great hospitality while I was working in his laboratory in Paris.

I would also like to acknowledge Amir Shapiro for his help with the robotic application. For your valuable help me in the lab, for the tremendous encouragement you gave me to develop the application and for the tools you provided in and out of the classroom.

Although they were not directly involved in the project I would like to express my appreciation to Valery Goussev who helped in the development of the wiener filter, and to Irena Chanis for her help and wisdom in the statistical analysis of the results.

I would also like to acknowledge Noam Peles for his help in the construction of the handles and bowls for the physical environment.

Special thanks goes to my dear friend Efi Vitzrabin, for your ideas, your patience and support in every possible hour of the day (including Skype talks from Paris, in the middle of the night), and for the countless hours we talked and debated on this thesis.

I would also like to thank my parents for the continuous support during the years, for introducing me to the pleasure of exploring and expanding my knowledge and for your morale support.

This work was partially supported by the Israel-France Research Networks Program in Neuroscience and Robotics.

Finally I would like to thank Daniela for the person she is for me, for her patience, morale support and love even though it meant letting me work in middle of the night, and leaving her behind for a month while I was working in Paris. Having her by my side made it a pleasant road to travel in.

Abstract

Constrained motion is a motion in which one or more of the natural degrees of freedom (DOF) of the system are restricted. Constrained motion is commonly encountered in various daily tasks such as: writing, drawing, or curving. Previous research of constrained motion, focused mainly on planar surfaces, however many of the constraining surfaces are curved e.g. bowls, plates, glasses, setting a different dynamic and kinematic problem. Hemispherical surfaces have constant curvature and form an interesting environment for constrained motion research. One of the main difficulties in exploring constrained motion is the construction of the constrained surface. Thus a virtual environment that allows complex surfaces alongside control of surface compliance is an important research tool. Yet virtual environments must be validated before they can be used for motion studies. Research of motion on curved surfaces extends our understanding of human motor planning and control. In the future it may also be used for applications such as development of objective motion measures and assistive robotics.

Several studies regarding non-constrained motion (pointing movements) and in drawing movements (on planar surface) found that gravity affected the kinematic characteristics of the movement. Differences in acceleration and velocity profiles of horizontal and vertical movement profiles were found. Several researchers have investigated the effects of gravity on motion planning in various setups, yet to the best of our knowledge, no research was done for non-planar constraint motion.

This research set to achieve three main goals: Validating the virtual environment through comparison of virtual motion profiles to motion profiles in a physical environment, studying the effects of gravity on hemispherical constrained motion and implementation of constrained motion algorithm based on hypothesized human strategies using a robotic system to facilitate improved understanding of underlying planning and control mechanisms.

To achieve those goals two human motion experiments were conducted, one in a virtual environment and another in a physical environment designed to be similar to the virtual environment. The virtual experiment included eleven subjects and a large set of movements in various directions on the virtual hemisphere. Horizontal and vertical movements were thoroughly analyzed for validating the virtual environment and for assessing the influence of gravity of motion planning and control in hemispherical constraint motion. Following the gained understanding from

the human experiments three types of hybrid control algorithms were developed and tested in the robotic experiment.

Results show that there are several differences in motion profiles between the physical and the virtual environments. Yet the differences can at large be ascribed by structure based differences between the environments and not to issues related to sensation feedback. Therefore the differences actually strengthen the validation of the virtual environment.

The results suggest that gravity effected motion in both environments. The effects found were reminiscing to effects found in previous research of gravity effect on motion of non-constrained motion (pointing movements) and drawing movements constrained to a planar surface. For example acceleration duration divided by the movement duration had a higher score in an up-to-down movement then in a down-to-up movement. Another interesting effect was that horizontal movements tended to dip downwards, below the straight-line projection connecting the targets while vertical movements were evenly distributed about the straight line. Those results suggest that the CNS may be taking gravity into account during motion planning.

The results of the robotic experiment validate that hybrid control is indeed suitable for constrained motion on a hemisphere. Due to limitations of hardware the results of the robotic and human experiment were only compared qualitatively, yet the results are encouraging as several similarities were found. The robotic movement profiles were very close to straight line projections, and the horizontal movements tended to dip downwards while the vertical movements did not. The later result suggests that the effects of gravity on motion profiles are, at partially due to inertial manipulator characteristics.

Keywords: constrained motion, hybrid control, human motor control, gravity, virtual environment.

Contents

CHAPTER ONE: INTRODUCTION	9
1.1 <i>PREFACE</i>	9
1.2 OBJECTIVES	10
1.3 IN SCOPE AND OUT OF SCOPE	11
1.4 MAIN CONTRIBUTIONS	12
1.5 THESIS STRUCTURE	12
CHAPTER TWO: LITERATURE REVIEW	13
2.1 GENERAL	13
2.2 ARM MOTION CONTROL	13
2.2.1 <i>Kinematics</i>	13
2.2.2 <i>Bang-Bang Control</i>	17
2.2.3 <i>Dynamics</i>	18
2.2.4 <i>Motion Control</i>	19
2.2.5 <i>Force Control</i>	20
2.2.6 <i>Combined force and motion control</i>	21
2.2.7 <i>Hybrid Control</i>	22
2.2.8 <i>Impedance Control</i>	23
2.3 HUMAN MOTOR CONTROL	25
2.3.1 <i>The Minimum Jerk Theorem (MJ)</i>	26
2.3.2 <i>The Minimum Torque Change (MTC)</i>	27
2.3.3 <i>Gravity effects on motion planning</i>	29
CHAPTER THREE: CONSTRAINT HUMAN ARM MOTION	30
3.1 PERFORMANCE MEASURES	30
3.3 VIRTUAL ENVIRONMENT - SUBJECTS AND EXPERIMENTAL PROCEDURE	33
3.4 PHYSICAL ENVIRONMENT - SUBJECTS AND EXPERIMENTAL PROCEDURE	35
3.5 PRE-PROCESSING	37
3.6 RESULTS AND STATISTICAL ANALYSIS	41
3.6.1 <i>Velocity Mean (VM)</i>	42
3.6.2 <i>Statistical analysis - AD_{total}/MD</i>	43
3.6.3 <i>Statistical analysis - AD_{rel}/MD</i>	45
3.6.4 <i>Statistical analysis - Path Straightness (PS)</i>	46
3.6.5 <i>Statistical analysis - Orthogonal bias (OB)</i>	48
3.7 DISCUSSION	51
CHAPTER FOUR: ROBOTIC CONSTRAINED MOTION	53
4.1 PERFORMANCE MEASURES FOR THE ROBOTIC EXPERIMENT	53
4.2 EXPERIMENTAL SETUP	54
4.3 POSITION BASED HYBRID CONTROL (PB-HC)	56
4.3.1 <i>Hybrid Vs Impedance Control for applying constrained motion</i>	57
4.4 THE HYBRID ALGORITHMS	58
4.4.1 <i>Hybrid control with Constant Stepping (CS)</i>	59

4.4.2 Hybrid control with Proportional (P) stepping.....	60
4.4.3 Hybrid control with Proportional Differential (PD) Stepping.....	61
4.5 RESULTS.....	62
4.5.1 PS and OB measures.....	62
4.5.2 ND, MD and FR Measures	64
4.6 DISCUSSION	65
CHAPTER FIVE: CONCLUSIONS AND FUTURE RESEARCH	67
REFERENCES	69
APPENDIX A – ELABORATION ON THE PSEUDO INVERSE KINEMATICS.....	73
APPENDIX B - PID CONTROL.....	74
APPENDIX C – PRE-PROCESSING – RECONSTRUCTION AND VALIDATION OF THE TIP	76
APPENDIX D- FULL POST HOC ANALYSIS - OB MEASURE FOR THE HORIZONTAL MOVEMENTS	79
APPENDIX E - SCOREBOT ER 5+ SPECIFICATIONS.....	81
APPENDIX F - FORWARD KINEMATICS AND JACOBIAN FOR SCORBOT ER 5+.....	83
APPENDIX G - INVERSE KINEMATICS FOR SCORBOT ER 5+	85
APPENDIX H - ATI MINI 45 – FORCE SENSOR SPECIFICATIONS.....	87
APPENDIX I - ROW DATA HUMAN EXPERIMENTS	88
APPENDIX J – ANALYZED DATA.....	89
APPENDIX K – RAW DATA ROBOTIC EXPERIMENT	90
APPENDIX L – PERCENTAGES ANALYSIS OF THE OUTLIER REMOVAL.....	91
APPENDIX M – README FILE	93
APPENDIX N – USER MANUAL FOR THE MATLAB SEGMENTATION	95
APPENDIX O - MATLAB CODES.....	98

List of Abbreviations

AD	Acceleration Duration
DB-IC	Dynamic Based Impedance control
CS	Constant Stepping
DOF	Degrees of freedom
DU	Down to Up, refers to the movement direction
FR	Forces Range
IM-IC	Instantaneous Model Impedance Control
LSD	Least Significant Difference
LR	Left to Right, refers to the movement direction
MD	Movement Duration
MJ	Minimum Jerk
MTC	Minimum Torque Change
ND	Number of Detachments from the hemisphere (robotic experiment)
OB	Orthogonal Bias
PB-HC	Position Based Hybrid Control
P	Proportional
PD	Proportional, Differential
PID	Proportional, Integral, Differential
PS	Path Straightness
Rel	Relvant
RL	Right to Left, refers to the movement direction
VM	Velocity Mean
UD	Up to Down, refers to the movement direction
$Trans_{z,d_i}$	Translation in Z axis of distance d_i
$Trans_{x,a_i}$	Translation in X axis of distance a_i
R_{z,θ_i}	Rotation around Z axis of angle θ_i
R_{x,α_i}	Rotation around X axis of angle α_i

List of Figures

Figure 1: Constrained motion in one direction, cleaning a table.	10
Figure 2: Two-link planar manipulator	15
Figure 3: Bang-bang control diagram.	17
Figure 4: Bang-bang control for a thermostat, operated by a human.	17
Figure 5: Block diagram of PID control loop.	75
Figure 6: The basic diagram of motion control.....	19
Figure 7: The combined control conflict, in a hemisphere.	21
Figure 8: Constrained task, writing on a board using hybrid control.	22
Figure 9: Min Jerk profiles using a MATLAB (Mathworks) simulation.	27
Figure 10: Normalized MTC profiles, (Engelbrecht and Fernandez, 1997).	28
Figure 11: OB calculation for a horizontal movement.	31
Figure 12: Plotting all the movements for subjects 1 (Blue) and 2 (Red).	32
Figure 13: The position of the virtual system compared to the subject.	33
Figure 14: A – The Delta 3 Haptic device. B – The computer screen.	33
Figure 15: Illustration of movements on the hemisphere.	33
Figure 16: Movement from the center target to the upward target in the virtual environment.	34
Figure 17: Target positions and movements on the physical sphere	35
Figure 18: Codamotion CX1 overhead unit.	36
Figure 19: Two physical hemispheres.	36
Figure 20: The handle apparatus.	37
Figure 21: Experimental setup, physical environment.	37
Figure 22: Wooden plate with three markers on back of the tip.....	38
Figure 23: Velocity, force and position profiles of a RL movement with segmentation.....	39
Figure 24: Velocity, force and position profiles of a RU movement without segmentation.	39
Figure 25: Error bar with 95% confidence level for the average VM measure.	43

Figure 26: MD plots for both environments.	43
Figure 27: Error bar with 95% confidence level for the average (AD_{total}/MD) measure.....	44
Figure 28: Error bar with 95% confidence level for the average (AD_{rel}/MD) measure.....	46
Figure 29: PS plots for both environments.	47
Figure 30: Error bar for horizontal and vertical movements, PS measure.	47
Figure 31: Average OB measure Vs the subject ID.....	49
Figure 32: OB plots for both environments. No units, ranges from -1 to 1.....	50
Figure 33: Error bar for horizontal and vertical movements, OB measure. Ranges [-1:1].....	50
Figure 34: Velocity profiles for a robotic and a human movement.	53
Figure 35: The apparatus attached to the manipulator.....	54
Figure 36: The axis in the experiment.	55
Figure 37: Position Based Hybrid Control, block diagram.....	56
Figure 38: Pseudo code illustration for the applied Hybrid control.....	58
Figure 39: Robotic position profiles for all hybrid algorithms in all the directions.	62
Figure 40: Robotic profiles - zoom in for LR and RL movements.....	63
Figure 41: Scorebot Er 5+ structure.....	65
Figure 42: The Codamotion axis.....	76
Figure 43: Defining the new transformation axes.....	76
Figure 44: Setup for testing the reconstruction.....	77
Figure 45: Reconstruction of a fourth marker.....	78
Figure 46: Plotting all the movements for one subject.	78
Figure 47: Scorebot Er 5+ structure, from the Intelitek Inc user manual	81
Figure 48: Scorebot Er 5+ engines, from the Intelitek Inc user manual	81
Figure 49: Overview on the gripper, from the Intelitek Inc user manual	82
Figure 50: Scorbot ER 5, with six degrees of freedom, from the Intelitek Inc user manual.	83

Figure 51: Axes for the calculation of engine 1 (m_1)	85
Figure 52: Axes for the calculation of engines 2 and 3 (m_2 , m_3).....	85
Figure 53: The axis of Ati Mini 45 force sensor.....	87
Figure 54: Opening screen of the Gui.....	95
Figure 55: The browse screen.	95
Figure 56: Force and velocity profiles on the Gui.	96
Figure 57: Do you wish to change marks and update segmentation question.	96
Figure 58: The addition of the position profile.	97
Figure 59: An option to correct the segmentation.....	97
Figure 60: An option to stop or continue.	97

List of Tables

Table 1: D-H parameters for 2-link planar manipulator.	15
Table 2: Experiment sets.....	34
Table 3: Outlier distribution of Horizontal and Vertical movements.	40
Table 4: The number of movements in each direction for every environment.....	41
Table 5: Three-way ANOVA results.	41
Table 6: VM two-way ANOVA.....	42
Table 7: LSD post hoc analysis for movement direction of the one-way ANOVA, VM measure.....	42
Table 8: AD _{total} /MD two-way ANOVA.....	43
Table 9: LSD post hoc analysis for movement direction of the one-way ANOVA,	44
Table 10: Between subjects test, PS measure for both environments.....	45
Table 11: LSD post hoc analysis for movement direction of the two-way ANOVA	45
Table 12: Between subjects test, PS measure for both environments.....	46
Table 13: LSD post hoc analysis for movement direction of the one-way ANOVA, PS measure.....	47
Table 14: Three-way ANOVA for vertical and horizontal directions separately.	48
Table 15: Partial LSD post hoc analysis for the OB measure, LR movement.....	49
Table 16: Partial post hoc analysis for the OB measure, RL movement.	49
Table 17: Hybrid control with CS, commands summary.....	59
Table 18: Hybrid control with P stepping, commands summary.....	60
Table 19: Hybrid control with PD stepping, commands summary.....	61
Table 20: PS measure, robotic experiment. Measure range [-1:1].....	62
Table 21: OB measure, robotic experiment. No units, ranges from -1 to 1.	63
Table 22: ND and MD measures – robotic experiment.	64
Table 23: FR measure – robotic experiment.	64
Table 24: Post hoc analysis – OB measure – LR movements	79
Table 25: Post hoc analysis – OB measure – LR movements, virtual environment.....	80

Table 26: Scorebot Er 5+ engines.....	81
Table 27: Acquired data from the virtual environment.....	88
Table 28: Analyzed data in the SPSS.....	89
Table 29: Analyzed data in the SPSS.....	90
Table 30: Outlier for all Movements, virtual environment.....	91
Table 31: Distribution of outliers per subject.	91
Table 32: Percentages distribution of outliers per subject.	91
Table 33: Outlier for all Movements, physical environment	92
Table 34: Distribution of outliers per subject.	92
Table 35: Distribution of outliers per subject.	92
Table 36: Percentages distribution of outliers per subject.	92

Chapter One: Introduction

1.1 Preface

Constrained motion refers to motion where one or more of the natural degrees of freedom (DOF) of the system are restricted. In a 3D space there are 6 DOF (3 linear and 3 rotational). Constrained motion is encountered in daily motion e.g. opening doors, driving, or playing an instrument. The door knob, the wheel, and the instrument (such as a violin) all constrain the hand to move in a specific path. Surface-constrained motion refers to motion along (and therefore constrained by) a surface. In the case of surface-constrained motion one linear and two rotation degrees of freedom are restricted leaving the system with 3 DOF (two linear and one rotational), e.g. when cleaning a table the surface of the table constrains the motion in one axis while the other axes are free (Figure 1).

One important issue regarding the research of constrained motion is the need to test hypothesis on various topographic shapes and compliances. However adapting the environment to different kinds of topographic shapes is expensive and time consuming. Thus a virtual environment that would allow creation of different curvatures with various compliances is an important research tool. However such an environment must be validated before it can be accounted for motion studies.

Previous research that examined motion on a hemispherical surface focused mainly on submovements and velocity profiles, the main finding was that such motions were characterized by bell-shaped velocity profiles similar to geodesic paths, (Sha et al., 2006; Liebermann et al., 2008). This thesis focuses on a different aspect which is testing the effects of gravity on constrained motion. Gravity affects were studied in unconstrained motion of whole body movements, sit to stand and back to sit, in vertical directions (Papaxanthis et al., 2003) and in pointing movements under gravity fields (Papaxanthis et al., 1998, 2003, 2005; Gentili, 2007). The effects of gravity on constrained motion was tested in drawing tasks on a planar surface, while this research deals with constrained motion on a hemisphere.



Figure 1: Constrained motion in one direction, cleaning a table.

Robotic constrained motion is a vast field of research. There are a multitude of application areas e.g., robotic surgery and robotic house cleaning, (Kemp et al., 2007). Robots may also be used for validating human motion studies. This approach is common in the field of cybernetics and a famous example of such use is "Walter's turtle" (Grey, 1950). Where two robots (*Elmer* and *Elsie*) were used to prove that complex behaviour can rise from a small number of brain cells with rich connections. Similarly in this research we program a robotic manipulator to validate hypotheses regarding constraint human motion.

1.2 Objectives

The high level objective of this research is to improve the understanding of motion planning and control of human hand movements. Constraint motion on a hemispherical surface poses a challenging motion planning problem and is thus an interesting research environment. In this research, constrained motion on a hemisphere in a virtual environment is tested and compared with a physical environment. Validation of the virtual environment is the first goal of the research. For this goal horizontal and vertical movements passing through the middle point of the hemispherical surface were analyzed. A second goal is to research the effect of gravity and surface constraint on arm motion. This is examined in both environments. A third goal is to lay the infrastructure for using robotic environment as a tool for human constrained motion. For this goal a robotic experiment was conducted in an environment similar to the physical environment.

1.3 In scope and out of scope

Human movements, in a virtual hemispherical-constrained environment, analyzed in this research were recorded as part of a research on "Mathematical modeling of 3D hand movements" funded by the ministry of science through the France-Israel research networks program in robotics and neuroscience. The experiment included horizontal, vertical, center out, and oblique movements of the hemispherical surface.

The research in this thesis focused on validation of the virtual environment by experiments conducted in a physical environment constructed to be as similar as possible to the virtual environment. A thorough kinematic comparison was conducted between movements in both environments. One of the major differences between the environments was the structure of the pointing device and the existence of a significant torque in the physical environment in comparison to a much smaller torque in the virtual environment. We therefore based the comparison on horizontal and vertical movements for which we expected gravity to affect movements differently in both environments. We did not analyze center out movements since initial screening showed they held less discriminating information, thus due to fatigue issues they were excluded from recordings in the physical environment. We pre-processed oblique movements but due to their complexity thorough analysis exceeded the scope of the current thesis. Since 6 Degree of freedom of endpoint force was recorded in the physical environment but only radial force was recorded in the virtual environment, comparison of force profiles proved futile.

In addition to human motion recordings in the physical environment we sought to test hypotheses regarding control algorithms applied by the CNS using a robotic system. Due to the limitations of the robot used this proved to be a very difficult task, facilitating only conceptual insights. In oblique movements some of the subjects used straight-line projections while others used curved movements. Using the robotic system we tested the hypothesis that a divided control methodology, i.e. separate control of the frontal plane and the sagittal plane (depth) would result in straight-line projections. For more in depth insights an improved robotic system must be developed. The infrastructure for such an implementation has been laid.

1.4 Main Contributions

The main contributions of this thesis are:

1. Differences were found between the physical and virtual environment. These differences point at the limitation of the virtual environment and suggest tuning analysis tools. Some of the differences between the environments may be attributed to the larger torque acting on the tool in the physical experiment and the different tool constraining mechanism and some perhaps to a difference in the dominant sensory feedback modality (force and vision). However since most of the differences had rational physical explanations those results strengthen the validation of the virtual environment, rather than weaken it.
2. Gravity effected constrained motion on the hemispherical surface. Acceleration duration (AD) divided by the movement duration (MD) was greater during a movement from the upper side of the hemisphere downwards (UD), compared to a movement from the down side of the hemisphere upwards (DU). Such asymmetry was not found in horizontal movements (left side to the right and vice versa). The results were repeated in both the physical and virtual environment (which meant difference in the speed execution and thus differences in the dynamics). These results suggest that the CNS takes gravity into account during motion planning.
3. Hybrid control is suitable for constrained motion on a hemispherical surface.

1.5 Thesis Structure

This thesis is organized as follows: chapter 2 presents the literature review. It includes a review on motion control and human motor control. Since this thesis is made of two main building blocks (human motor control and robotic motion control) each is described and analyzed separately (chapters 3 and 4 respectively). Chapter 3 includes a full description of the experiments which were done with human subjects in the virtual and the real physical environment while chapter 4 includes the description and analysis of the robotic experiment. Finally chapter 5 combines the two threads and presents conclusions and future research directions.

Chapter Two: Literature Review

2.1 General

This chapter begins with kinematics, the mathematical basis for human and robotic motion and following motion control is reviewed. This section includes basic control methods such as bang-bang followed by basic control inputs which are commonly used, e.g. position, velocity, acceleration and Force control. The section ends with a review on how to combine force and position control using hybrid or impedance control.

Relevant issues in human motor control are discussed in section 2.3. Topics covered include path planning, gravity effects, and motion characteristics. It is commonly assumed that the motor system tries to optimize a certain function related to the execution of the movements. Such an optimization can be based on kinematic or dynamic features of the planned trajectory. In this thesis we review the Minimum Jerk (MJ) kinematic model and the Minimum Torque Change (MTC) dynamic model. Finally studies regarding effects of gravity on constrained and non-constrained motion are presented.

2.2 Arm Motion Control

2.2.1 Kinematics

Kinematics describes motion without consideration of related forces and torques. The kinematic description is therefore a geometric one and its basic block buildings are forward and inverse kinematics. Forward kinematics is the problem of locating the position and orientation of the end link, given the joint angles (Kucuk and Bingul, 2006). Inverse kinematics solves the reverse problem which is given the position of the end link what are the corresponding joint angles.

2.2.1.1 Forward kinematics

Forward kinematics is concerned with the relationship between the individual joints of the robot manipulator and the position and orientation of the end-effector (Spong, 2007). In this review it is assumed that all joints have a single degree of freedom like in the case of revolute or prismatic joints (angle of rotation and linear displacement respectively).

A robot manipulator with n joints will have $n+1$ links, since each joint connects two links. Calculation of the end effector position involves a kinematic chain of n serial links, and in a mathematical expression it will be a multiplication of the matrixes that represent each joint. For each joint we will write the corresponding axes system and by using the Denavit-Hartenberg (D-H) convention we would create the needed transformation. The D-H has four basic parameters a_i , θ_i , d_i , and α_i , termed: link length, link twist, link offset, and joint angle, respectively. Each homogeneous transformation A_i is represented as a product of four basic transformations:

$$\begin{aligned}
 A_i &= R_{z,\theta_i} \cdot Trans_{z,d_i} \cdot Trans_{x,a_i} \cdot R_{x,\alpha_i} = \\
 (2.1) \quad &= \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & \alpha_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) & 0 \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \\
 &= \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) & a_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\alpha_i)\sin(\alpha_i) & a_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}$$

Once the four basic D-H parameters are determined for each joint, the position and orientation of the end effector (for n links) are given by:

$$(2.2) \quad H = A_1(q_1) \dots A_n(q_n)$$

Figure 2 includes an example of using forward kinematics for solving a two linked planar manipulator (Spong, 2007). The joint axes, of the two linked planar manipulator, z_0 and z_1 point out of the page and are not shown in the Figure 2, meaning they are normal to the page. The origin is chosen at the point of intersection of the z_0 axis with the page and the direction of the x_0 axis is completely arbitrary. The final frame is fixed at the end of link 2.

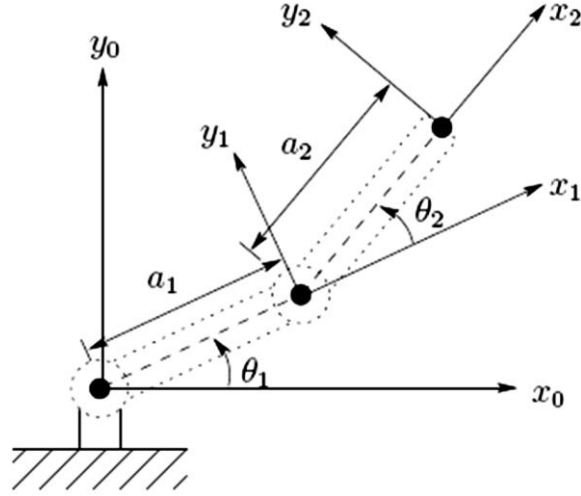


Figure 2: Two-link planar manipulator

The manipulator's D-H parameters are given in Table 1.

Table 1: D-H parameters for 2-link planar manipulator.

Link	a_i	α_i	d_i	θ_i
1	a_1	0	0	θ_1
2	a_2	0	0	θ_2

a_1 and a_2 are link lengths, θ_1, θ_2 are joint rotation angles.

Using equation (2.1) the following homogenous matrix is applied:

$$(2.3) \quad A_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(0) & \sin(\theta_i)\sin(0) & a_i \cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(0) & -\cos(\theta_i)\sin(0) & a_i \sin(\theta_i) \\ 0 & \sin(0) & \cos(0) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & a_i \cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) & 0 & a_i \sin(\theta_i) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The final configuration is given by:

$$(2.4) \quad H = A_1 \cdot A_2$$

2.2.1.2 Inverse kinematics

Inverse kinematics determines the values of the joint variables when given the end-effector position and orientation. The inverse function is $f^{-1} = (d_1 \dots d_n, \theta_1 \dots \theta_n)$. In principal there are two basic ways for solving inverse kinematics: analytic and iterative methods (which sometimes make use of numerical computations since there is no analytical solution). For Example 1, the two-link planar manipulator obtaining an analytical solution is straight forward (equation 2.5 and 2.6), But, for the general case obtaining an analytical solution may be very difficult.

$$(2.5) \quad \cos(\theta_2) = \frac{x^2 + y^2 - \alpha_1^2 - \alpha_2^2}{2 \cdot \alpha_1 \cdot \alpha_2} = D \quad ; \quad \theta_2 = \cos^{-1}(D)$$

$$(2.6) \quad \theta_1 = \tan^{-1}\left(\frac{x}{y}\right) - \tan^{-1}\left(\frac{a_2 \sin(\theta_2)}{a_1 + a_2 \cos(\theta_2)}\right)$$

There are many different iterative methods for constructing the inverse kinematics: for example the Jacobian transpose, the Pseudo Inverse, the DLS - Damped Least Squares (Wapler, 1986), the Cyclic Coordinate Descent method, Quasi Newton and Conjugate Gradient method (Buss, 2004; Wang and Chan, 1991).

For a k DOF manipulator lets define the current position of each joint as q_i ($i=1 \dots k$) and the target position as t_i ($i=1 \dots k$) we can write the error which is the desired change in the position as: $e_i = t_i - q_i$. The Jacobian matrix is defined as:

$$(2.7) \quad J(q, \theta) = \begin{bmatrix} \frac{\partial q_1}{\partial \theta_1} & \frac{\partial q_1}{\partial \theta_2} & \dots & \frac{\partial q_1}{\partial \theta_m} \\ \frac{\partial q_2}{\partial \theta_1} & \frac{\partial q_2}{\partial \theta_2} & \dots & \frac{\partial q_2}{\partial \theta_m} \\ \vdots & \vdots & & \vdots \\ \frac{\partial q_n}{\partial \theta_1} & \frac{\partial q_n}{\partial \theta_2} & \dots & \frac{\partial q_n}{\partial \theta_m} \end{bmatrix}$$

Where θ is a vector of the joint angles and which defines the position of the joint. Now the basic equation can be written as:

$$(2.8) \quad \dot{q} = J(\theta) \dot{\theta} \rightarrow \Delta q \approx J(\theta) \Delta \theta$$

The question is how to choose $\Delta \theta$ and the main idea is to choose $\Delta \theta$ so that Δq will be close as possible to \vec{e} . More elaboration on the Pseudo inverse appears in Appendix A.

2.2.2 Bang-Bang Control

Bang-bang control is a discrete control method which switches between two (or more) states according to a certain threshold, Figure 3. They are often used for systems which require a binary input and today most of the thermostats are based on bang-bang controllers, and the most simple example is an the operation of an air condition by human, Figure 4. It is also very common to find the bang-bang control in light operations, pump fuelling, etc.

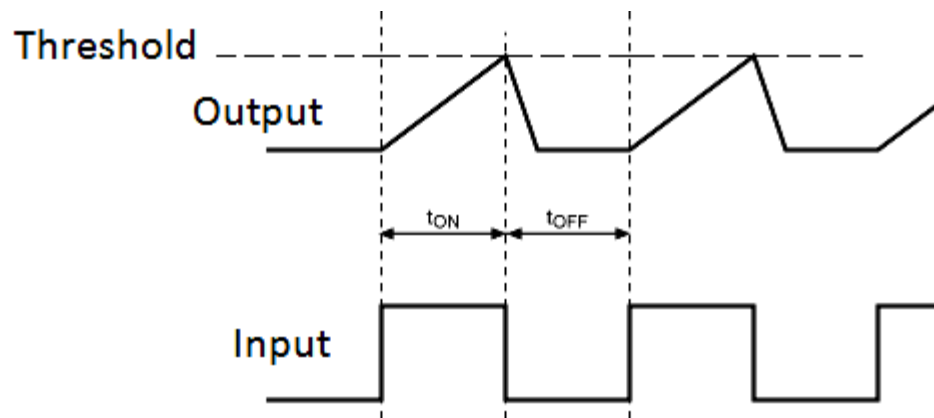


Figure 3: Bang-bang control diagram.

The input "jumps" between states according to the output of the system.

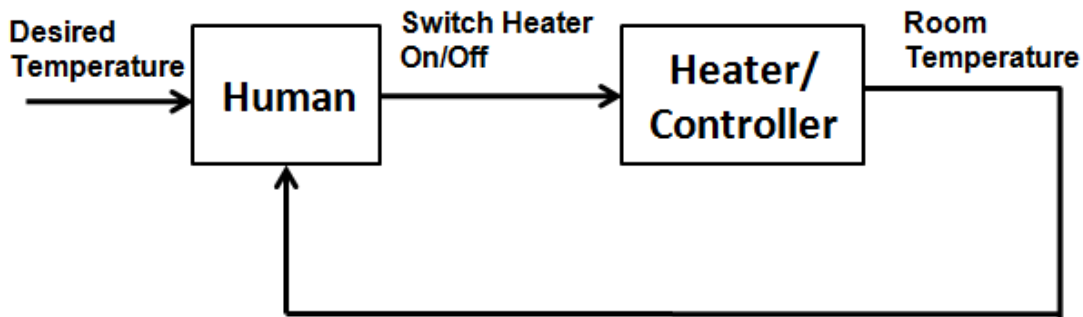


Figure 4: Bang-bang control for a thermostat, operated by a human.

Bang-bang control has the advantage of algorithmic simplicity on one the hand, and on the other hand it was proven to be a sufficient control method for minimum time problems. The criteria for minimum time problems, is to reach a certain goal in the shortest time. The goal could be driving or moving from point to point, heating and so on. The solution for the minimum time problems is usually described as a maximum acceleration of the system until it reaches the switch point and then a maximum deceleration. Bang-bang control provides such solutions in the most suitable way.

2.2.3 Dynamics

While kinematics describes the motion, dynamics describes the causes for the motion, in other words dynamics is concerned with the relationship between the forces acting on a mechanism and the motion (accelerations) they produce. Similar to the field of kinematics, dynamics is often divided in two main fields:

- Forward dynamics: given the forces – describe the motion (accelerations).
- Inverse dynamics: given the motion (accelerations) – describe the forces.

A common way for describing the dynamical model of rigid robot manipulators is by using Lagrange dynamics. Let the robot manipulator have n links and let the $(n \times 1)$ -vector q of joint variables be $q = [q_1, \dots, q_n]$. The corresponding dynamic model is:

$$(2.9) \quad M(q) \cdot \ddot{q} + C(q, \dot{q}) \cdot \dot{q} + G(q) = \tau$$

Where $G(q)$ is the $(n \times 1)$ -vector of gravity force, $C(q, \dot{q})$ is the $(n \times 1)$ -vector of Coriolis and centrifugal forces, $M(q)$ is the $(n \times n)$ inertia matrix, and τ is the $(n \times 1)$ -vector of joint control inputs to be designed. Friction and disturbance input have been neglected here.

Some important properties of M , C and G are:

- 1) The inertia matrix is a symmetric positive-definite matrix, all of its eigenvalues are positive and thus for every vector $\vec{V} \neq 0$ it holds that $\vec{V}^T \cdot m \cdot \vec{V} > 0$.
- 2) It is possible to choose $C(q, \dot{q})$ so that the corresponding matrix $R = \dot{M} - 2C$ is skew symmetric, which means that for every selection of C the following property consists $\dot{q}^T (\dot{M} - 2C) \dot{q} = 0$.
- 3) There is always a certain constant upper bound (g_0) for the gravity force/torque which satisfies: $\|G(q)\| \leq g_0$

2.2.4 Motion Control

Motion control is the problem of tracking a desired trajectory by determining the necessary control inputs (Spong, 2007). The control inputs could be the position, velocity, acceleration or force and the common control is the PID control (for more see Appendix B).

The main goal is to design a feedback controller such that the joint coordinates $q_r(t) \in R^n$ track the desired motion $q_d(t)$ as closely as possible. The basic concept of motion control also referred to as joint space control is shown in Figure 5. The desired motion (X_d), which is described in terms of end-effector coordinates, is converted to joint trajectory (q_d) using inverse kinematics of the manipulator. Then the feedback controller which is affected by the current (q_r) and the desired (q_d) trajectories and states, defines the torque (τ) commands for the manipulator.

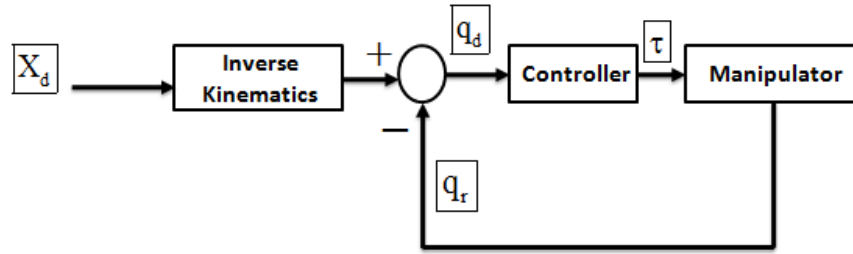


Figure 5: The basic diagram of motion control.

X_d is the desired position, q_d and q_r are the corresponding joint trajectories and joint states.

Position control is the most commonly control method used for robots applications (Spong, 2007). The goal of the control method is to bring the robot to a specific position in space, thus to bring the position error between the desired and current position to zero. The natural use of position control is when there are no forces which affect the mission, or the forces are not part of the considerations and the most classical example is in the field of robot navigation.

Similar to the position control there is the velocity control, in which we control the velocity in order to achieve an accurate position. By using velocity control, we can control the movement velocity and by doing so we increase our accuracy (reducing the speed) if needed or decrease it if not needed.

Acceleration control requires a complete dynamic model of the controlled system and is thus more complicated to implement. This method was developed in order to improve the position accuracy and to control the rate of change in the velocity in order to avoid drastic changes in the movement. Thus the movements which will be created will be smoother.

2.2.5 Force Control

Force control is needed for tasks which require controlling on endpoint forces e.g. gluing, gripping or fine object manipulations. The goal of the control system is to minimize the difference between the actual and target forces. A gluing task is a good and pure example for force control since this action doesn't require movement. There is only need for a stable pressure where low pressure might not get the job done (a partial gluing result) and strong pressure might damage the object. There are two basic methods used for force control: passive and active.

Passive force control is an open loop control, while compliance of the robot's end-effector allows it to adjust to the interaction forces. The passive approach to interaction control is very simple and cheap, because it does not require force/torque sensors, however it lacks flexibility and since no forces are measured it cannot ensure that high forces would not accrue.

Active force control is a closed loop control system that can adjust to reduce the force errors. It uses measurements of the force in order to ensure the stability of the forces (in the limited range) and to avoid strong forces. This method is more expensive, slower and complex due to the force sensors which need to be involved. However it has more flexibility and can be used for a variety of tasks. In order to obtain a reasonable task execution speed the active force control must have a certain degree of compliance, since the feedback and error has to accrue after a motion started, which means that without the compliance the reaction forces might cross the undesired threshold.

2.2.6 Combined force and motion control

Surface constrained tasks require a combination of force and position control. Using a combined control method may result in a conflict in the commands which are given to the robotic engines. A good example that describes the conflict is shown in Figure 6. The given task is to move from point A to point B, while maintaining contact with the hemisphere during the entire movement. In this case the position controller would want to move straight to the final target in lowest cost for the engines and would not maintain contact during the movement. The force controller would want to change the direction of the movement downwards or maybe into to hemisphere in order to maintain contact. As a result there might be a conflict with the commands which are given to the motor engines of the robot. This duplication of commands is known as the integrated control conflict and there are several methods to solve this conflict such as *impedance control*, *hybrid control* (active control) and others which use a passive control manner. In this research we focus on active control and specifically we use hybrid control as the selected algorithm

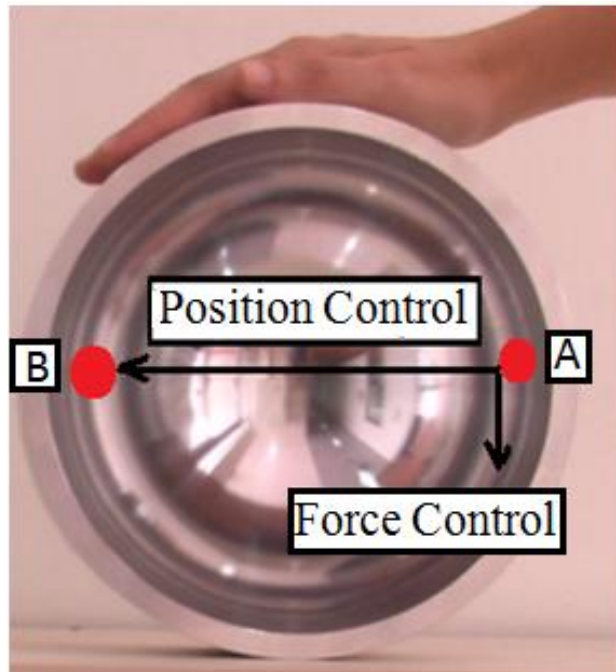


Figure 6: The combined control conflict, in a hemisphere.

The position control wants to move from A to B, while the force might want to move downwards to maintain contact.

2.2.7 Hybrid Control

Hybrid control is an active control method which uses force and position control in parallel. The force and position control can be used independently only if they remain orthogonal to each other during the entire movement. Writing on a planar board is classical for hybrid control (Figure 7) since the position vector is limited to the surface, axis X - Y, and the force vector is directed into the surface, axis Z. Thus they are orthogonal at all time.

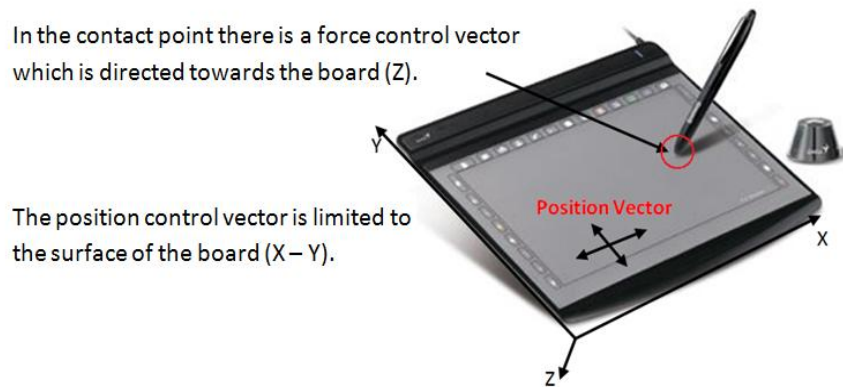


Figure 7: Constrained task, writing on a board using hybrid control.

The original hybrid control concept was introduced by Raibert and Craig, 1981. Their method used only kinematics and did not take into account the manipulators dynamics as part of the control. Yoshikawa et al. (1987, 1988) was the first to propose a basic structure for dynamic hybrid controller and was followed by Hogan and Colgate (1989) which also investigated the instability of the method. It was shown that when a manipulator with 6 or more degrees of freedom is used, the effector position and force can simultaneously be controlled as long as the manipulator does not fall into singular configuration, Hogan and Colgate (1989).

Approaches designed to cope with uncertainties in the environment geometry are the force control with feed forward motion scheme proposed in (Schutter and Brussel, 1987) and the parallel force/position control (Chiaverini and Sciavicco, 1993), based on the concept of dominance of force control on motion control, thus the force control dealt with the uncertainty during the entire action and position control was satisfied only if possible. Some suggested using vision in order to overcome the uncertainties in the environment. The benefits of combining vision in the feedback loop of a robot manipulator were first presented by Nelson et al. (1995) which suggested that different directions of the task space are simultaneously controlled using vision and force sensors. The advantages of using visual servoing as part of the control scheme are clear and beyond this work, however there were also few who showed that contact stability can be achieved by estimating the constraint geometry by using

only feedback from the force sensor and the and the position feedback from the engines (Namvar and Aghili, 2004; Karayiannidis and Doulgeri, 2006).

Antonio et al., (2009) proposed a hybrid vision–force control method for the visual tracking of a desired trajectory, while the camera parameters, as well as the constraint surface, were assumed to be uncertain. The proposed scheme was applied for motion of a manipulator on a cylindrical surface.

2.2.8 Impedance Control

Impedance control was inspired by circuit theory where the impedance is defined as the ratio between voltage and current in the circuit. The analogy is that instead of controlling each of the position and force separately it is possible to control the ratio between them (or the ratio between the force and the derivatives of the position (velocity, acceleration...)).

In mechanical systems the impedance is the ratio between the force and the position vector. The force results from the interaction forces between the robots (usually at the end point) to the environment. The position vector is defined between the actual position, velocity or acceleration of the end point compared to the initial position. The literature is filled with many algorithms for implementing impedance control. In the following sentences three different and complementing ways are reviewed.

Dynamic Based Impedance control (DB-IC) - The main idea is that using a correct force in the robot's engines we can determine the impedance at the end point or any other desired point via feedback from the observed environment. The dynamic impedance control equation is:

$$(2.10) \quad F_{\text{int}} = K(\Delta x) + B(\Delta \dot{x}) + M(\Delta \ddot{x})$$

The position error (Δx) is derived from the connection between the desired position X_D and the current position, X_C . $\Delta x = x_D - x_C$. K is the stiffness, B is the restraining matrix and M is the inertia matrix. The impedance is actually the proportion between the interaction forces and the position error.

The Dynamic Based Impedance control (DB-IC) method was first developed by Colgate and Hogan (1988). Many of the researchers followed this method and used it for a variety of different tasks which requires interaction with the environment. Application of the DB-IC can be found in the literature from learning applications with manipulators (Cheah and Wang, 1998), combined tasks with multiple robots (Bonitz and Hsia, 1996) and many more.

Position Based Impedance Control (PB-IC) – The main principal of the method is to track the impedance that is changing according to the interaction force in time. This model creates a new

position vector, and the robot needs to follow the model instead of the position or the original force. The model is created for a dynamic system (linear equations of a second degree), and the calculations are made by numerical computation methods.

This method was first published by (Lawrence, and Stoughton, 1987) who later on tested its stability properties (Lawrence, 1988). It was shown to be suitable for physical assignments such as cooperation task for human and robot (Ikeura and Inooka, 1995) and suitable for use in industrial robots (Šurdilovic and Kirchhof, 1996; Šurdilovic, 1998).

Instantaneous Model Impedance Control (IM-IC) - The Instantaneous model Impedance Control (IM-IC) was suggested in order to overcome the limitations of the first two methods (Valency and Zacksenhouse, 2000; 2003). The IM-IC re-initializes the impedance model to the current position of the robot so the model does not accumulate position errors. The model has the advantage of using position feedback both in the outer loop, to track the desired impedance, and in the inner to improve the robustness.

The model is designed to take advantage of the error-correction capabilities of position controllers, such as the PB-IC, while maintaining good impedance tracking performance, like the dynamic based impedance controller (DB-IC). Moreover, this method facilitates the trade-off between accurate impedance tracking and robust performance. It calculates the acceleration needed to fulfill the impedance relationship as in the DB-IC approach. Using the acceleration and the current position and velocity, it predicts the position of an instantaneous model of the desired impedance in the next time step. The predicted position is converted to the joint space and used as a reference to the inner position controller. The inner controller can be any position controller that assures stable trajectory tracking.

The *DB-IC* method main disadvantage is that most of the industrial controllers are position based and thus it is difficult to use in many cases. The *PB-IC* method is position based and thus it is compatible for the industrial controller. However, its performance in tracking the impedance error (compared to the dynamic model), is worse and another disadvantage is the fact that this method uses inverse kinematic equations which sometimes include un invertible matrices. The IM-IC method used the advantages of both methods, but still has a large disadvantage for applying constrained motion. This method tracks the relation between the force and position control (impedance) and cannot track a specified path or a specified set of forces.

2.3 Human motor control

In order to understand why the brain chooses a particular motor plan among the infinite possibilities it is commonly assumed that the motor system tries to optimize a certain function related to the creation of the movements. Flash and Hogan (1985) addressed the path planning optimization question and suggested the Minimum Jerk (MJ) optimization model. According to the MJ model, humans attempt to maximize smoothness by minimizing the rate of change in acceleration, i.e. jerk. On a planar surface the model predicts straight-line paths and bell shaped speed profiles. Experiments showed that human reaching movements are characterized by smooth bell-shaped velocity profile (and thus suitable for the predication made by the MJ model), and a small portion of irregularities at the end of the movement. Many researches referred to it as a primary submovement and a secondary corrective submovement or a third and so on (Meyer et al., 1988; Navak et al., 2000; Elliott et al., 2001; Ketcham et al., 2002; Dounskaia et al.).

Sha et al. (2006) suggested that the trajectory of minimum jerk movements on the surface of sphere should be along great circles that are the geodesic in this environment. Yet experimental results do not completely support this notion as path did not converge to the great circles even after practice, Sha et al. (2006). Similar results in a different experimental environment were obtained by Liebermann et al., (2008). Our work addresses the issue of straight-line paths, but not the issue of velocity shape profile.

The MJ model was formulated in the mid 80's and was consistent with most of the experimental data available at the time (Engelbrecht and Fernandez, 1997). However, various later findings, revealed discrepancies between minimum-jerk trajectories and observed behaviour. In particular, there was increasing evidence that hand paths are curved for certain point-to-point reaching tasks and that hand-path curvature systematically varies with workspace location (Atkeson and Hollerbach 1985; Flanagan and Ostry 1990; Flash 1987; Kaminsky and Gentile 1986; Lacquaniti et al., 1986). Motivated by the MJ model discrepancies, Uno et al. (1989) proposed The Minimum Torque Change (MTC) model. While the MJ model is purely kinematic the MTC is a dynamic model for path planning. The MTC basic concept is the time integral of the squared sum of torque changes is minimal. This proposition initiated a continuing controversy between the supporters of the minimum-jerk and the minimum-torque-change principles (Kawato 1996). However, both models were quite similar in their temporal prediction (Engelbrecht and Fernandez, 1997). Without arguing in favour or against the MTC and MJ model, several studies on the effect of gravity on motion suggested that during vertical arm

movements, the brain optimizes the inertial (speed dependent) force more than the gravity force acting on the arm (Soechting et al. 1995; Soechting and Flanders 1998). It was also shown that that when movement speed decreased (in vertical movements) subjects guided these movements by increasing or decreasing the muscular force of the flexor muscles only and thus suggested that CNS exploits the gravitational force for optimizing the overall muscle activation (Papaxanthis, 1998). How gravity effects constrained motion is still an open question, but it is clear that it plays an important role in the motor plan.

2.3.1 The Minimum Jerk Theorem (MJ)

In the Minimum Jerk model (Flash and Hogan, 1985) the rate of change in the acceleration (jerk) is minimized. This is formulated in the following cost function:

$$(2.11) \quad C_j = \frac{1}{2} \int_0^{\tau_f} \left[\left(\frac{d^3x}{dt^3} \right)^2 + \left(\frac{d^3y}{dt^3} \right)^2 \right] dt$$

Where τ_f is the movement duration, x and y are the actual movement in space.

The model predicts straight paths and bell shaped velocity profiles, often observed in human reaching movements. Experimental results (Flash and Hogan, 1985) show a high correlation between planar human hand motion and the predictions of the MJ model. The fundamental equation which was obtained from the optimization is:

$$(2.12) \quad x(t) = x_0 + (x_f - x_0) \cdot (15 \cdot \tau - 6 \cdot \tau^5 - 10 \cdot \tau^3)$$

Where x_0 , and x_f , define the start and end point of the movement respectively. The total duration time of the movement is $(t_f - t_0)$ and $\tau = t / (t_f - t_0)$. An illustration of the MJ theorem is shown in Figure 8.

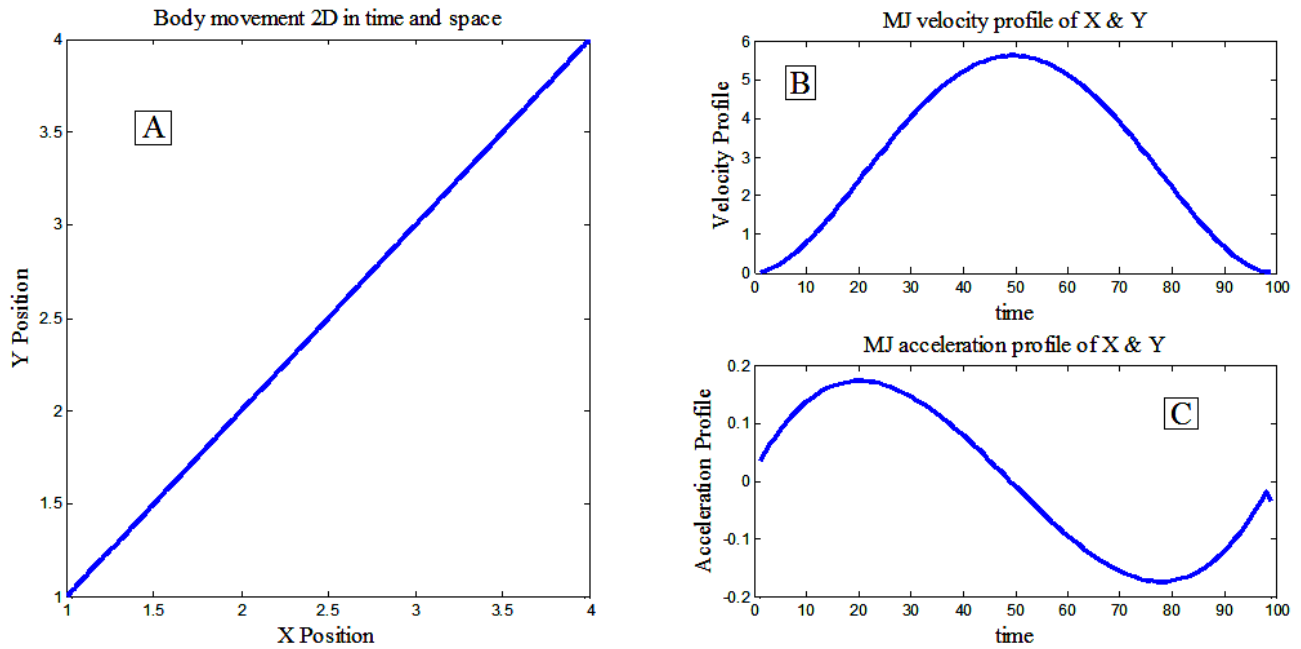


Figure 8: Min Jerk profiles using a MATLAB (Mathworks) simulation.
A - Movement on an X-Y plane. B – The velocity profile. C – The acceleration profile.

2.3.2 The Minimum Torque Change (MTC)

The MTC optimization is a minimization of the torque change and its by-product is smooth muscle torques, (Uno et al., 1989). While the early research and results were obtained by use of numerical methods (Uno et al., 1989). Engelbrecht and Fernandez (1997) showed that for in special cases of one-DOF planar-horizontal movements, minimum-torque-change trajectories may be found analytically. They showed that for these movements the MTC trajectories apply several properties that persist for any set of arm dynamics and any movement duration:

- 1) A symmetric velocity profile.
- 2) The ratio of peak velocity to mean velocity is bigger than $3/2$ and smaller than $15/8$.
- 3) MTC trajectories have a single velocity maximum which is always located at the movement time midpoint.
- 4) The first peak of any MTC acceleration profile must occur before 21.13% of the total movement time.

However, those properties hold only for certain types of movements and are not universal. Several researches showed evidence that velocity profiles of slow movements are generally right-skewed (Moore and Marteniuk, 1986; Nagasaki, 1989), while fast movements are left skewed

(Wiegner and Wierzbicka, 1992) their ratios of velocity peak to average tend to exceed 15/8 and their acceleration peaks may occur significantly later than 21.13% of total movement duration (Baba and Marteniuk, 1983). The cost function of the MTC model is defined as an integration of the torque changes over the movement duration $[0, t_f]$:

$$(2.13) \quad C_{MTC} = \frac{1}{2} \int_0^{\tau_f} \dot{\tau}^2(t) dt$$

τ is the muscle torque. For the optimization the Euler-Lagrange equation must be satisfied:

$$(2.14) \quad \frac{d^3}{dt^3} \cdot \frac{\partial \dot{\tau}^2}{\partial \ddot{\theta}} - \frac{d^2}{dt^2} \cdot \frac{\partial \dot{\tau}^2}{\partial \dot{\theta}} = 0$$

The fundamental equation (normalized for enabling comparisons between trajectories of various durations) obtained from the optimization is:

$$(2.15) \quad \theta(\tau) = \theta_f \frac{3}{w^2 - 3w \coth(w) + 3} \cdot \left(\frac{\sinh(w\tau)}{\sinh(w)} - \frac{1}{6} w^2 \tau^3 + \left(\frac{1}{2} w^2 - w \coth(w) \right) \cdot \tau \right)$$

τ is defined as the time (t) divided by the total duration time (T).

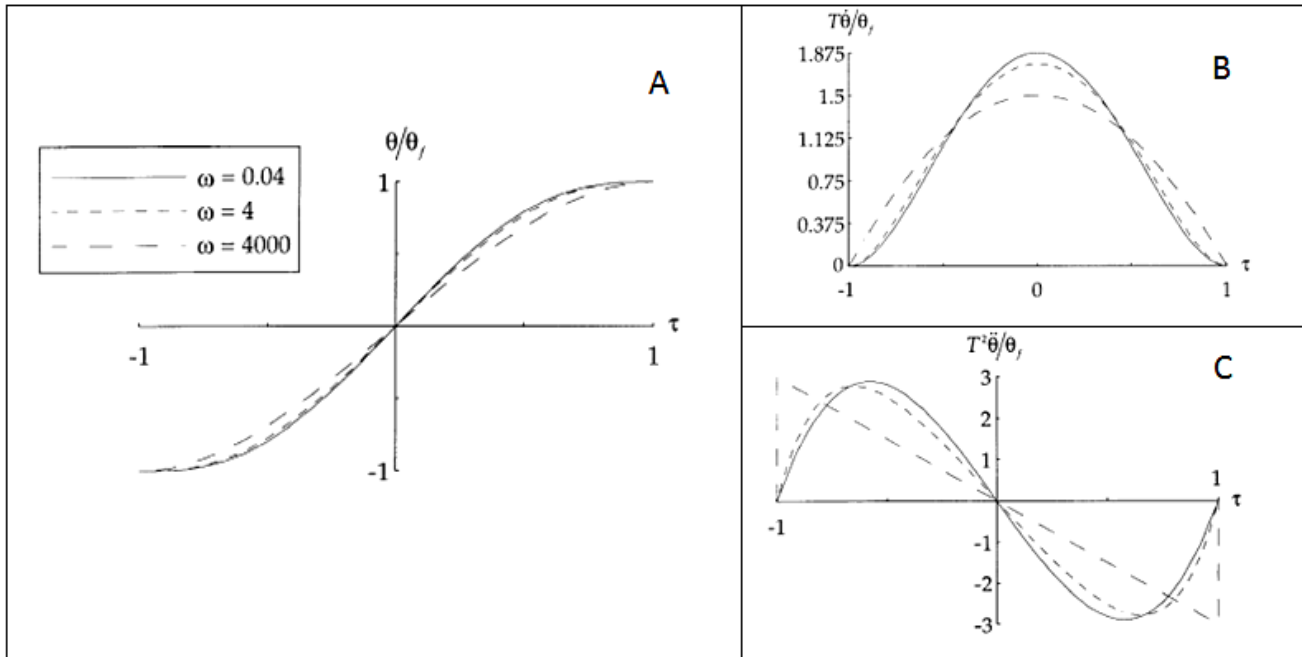


Figure 9: Normalized MTC profiles, (Engelbrecht and Fernandez, 1997).

Time and amplitude normalized MTC trajectories (A), velocity profiles (B), and acceleration profiles (C) for $w=0.04$, 4, and 4000.

2.3.3 Gravity effects on motion planning

The effect of gravity on human motor control was researched in several different contexts e.g. whole body movements (Papaxanthis et al., 2003), drawing movements (Papaxanthis et al., 1998), pointing movements (Papaxanthis et al., 2005) and in gravity fields (Gentili et al., 2007).

Papaxanthis et al., (1998, 2003) report that hand kinematics in pointing and drawing tasks vary when the direction of the movement changes, e.g. acceleration duration (AD) divided by the movement duration (MD) is greater during a downward (UD) movement compared to an upward (DU) movement. Those differences did not appear in horizontal movements, i.e. left-to-right (LR) and right-to-left (RL) movements. Such asymmetries were also found in vertical whole body movements (Papaxanthis et al., 2003). Results show that in sit to stand and back to sit movements AD divided by MD were greater during UD movements while compared to DU movements.

Soechting et al. (1995) found that optimisation principles based on inertia also predict different paths for UD and DU movements and thus the direction-dependent differences in the kinematic characterization of the movements are not the result of only the gravitational effects. A recent research on movement direction, used different speeds of pointing movements to test the effect of dynamics (inertial and gravity torques change differentially with movement speed) (Papaxanthis et al., 2003). The results showed that differences between the movement directions (UD Vs DU) continued to exist although motion dynamics changed. These results seem to contradict the previous hypothesis, and thus Papaxanthis et al. (2003) hypothesised that the differences are robust and cannot be attributed only to motion-mechanics or muscular-mechanics. To further address this question, experiments in microgravity expanded the view on this issue and showed that in the zero gravity case, the kinematic differences were reduced when exposure time to microgravity increased, and that the velocity profiles became more and more symmetric. These results led the authors to a conclusion that the planning and execution of human arm movements are affected by the dynamics and the kinematics of the motion (Papaxanthis et al. 2005). Gentili et al. (2007) proposed that the asymmetries in the motion characteristics could be due to a minimization criterion which compromise between the absolute work of external forces and a comfort term. To the best of our knowledge research on gravity effects in constrained motion control was conducted on planar surfaces only. Movements constrained to a hemisphere form a different kinematic and dynamic problem. Thus we set out to investigate the effects of gravity on motion planning in such an environment.

Chapter Three: Constraint human arm motion

Constrained human arm movements on a hemispherical surface were recorded in two different environments. One experiment was conducted in a virtual environment and the other in a physical environment. The objective of the virtual experiment was exploring constraint motion. In this thesis we specifically concentrate on exploring the effects of gravity on surface constraint motion. Although several types of movements were recorded in the experiment, only vertical (DU and UD) and horizontal (LR and RL) movements are examined for testing the effects of gravity on constrained motion and to validate the virtual environment. The main objective of the experiment in the physical environment is for validating the virtual environment.

This chapter begins with detailing the performance measures, followed by the description of the different experiments and environments (virtual and physical) in section 3.3 and 3.4. The results and the statistical analysis are given in section 3.5 and a discussion on the results is given in the final section.

3.1 Performance Measures

Three measures for comparing the environments and two measures for assessing the effects of gravity were defined. The measures for comparing the virtual and the physical environment are:

Velocity Mean (VM) – VM is calculated from the velocity profile after segmentation. It was calculated by dividing the measured path length with the total movement time.

Movement Duration (MD) – MD is defined as the time between the movement onset and the movement offset.

Path Straightness (PS) – PS is defined as the Euclidean distance between the start and end points, divided by the actual path length. The Euclidean distance is calculated by

$$\sqrt{(Y_{\text{Start}} - Y_{\text{End}})^2 + (Z_{\text{Start}} - Z_{\text{End}})^2} . \quad \text{The actual path length was calculated by}$$
$$\sum_{K=\text{start}}^{\text{end}} \sqrt{(Y_{K+1} - Y_K)^2 + (Z_{K+1} - Z_K)^2} .$$

The measures for assessing the effects of gravity on constrained motion on a hemisphere are:

AD/MD – Acceleration duration (AD) is defined as the time between the movement onset and the velocity peak normalized by MD (movement duration). Previous research reported that asymmetry of the AD measure can be found in vertical movements for which AD was greater during a UD movement compared to a DU movement. However, when comparing LR and RL movement the

asymmetry was not found (Papaxanthis et al., 1998, 2003). When testing AD it is important to determine how the acceleration is calculated. Two ways for calculating the acceleration were tested:

- 1) $AD_{total} = \sqrt{a_x^2 + a_y^2 + a_z^2}$ and thus would be referenced as AD_{total}/MD .
- 2) AD_{rel} – is the acceleration only from the main axis of the movement e.g. for horizontal movements the movement was mainly along Z axis and thus only the acceleration in Z is taken into account, meaning $AD=a_z$. In vertical movements, the movement is along Y axis and thus $AD=a_y$ and thus "rel" refers to relevant.

Orthogonal bias (OB) – OB is measured as the difference between negative and positive mean-square-error (MSE) from the line projection on the hemisphere, divided by their sum. The negative and the positive MSE were found by calculating the integral between every 2 points of the actual movement compared to the projection of a straight-line (Figure 10). For example in Figure 10 the blue line refers to the movement and the black line refers to the projection of a straight line. The integral was calculated between difference of the blue line and the black line. If the integral between the two points is negative it is termed negative mean square error, else it is termed positive mean square error.



Figure 10: OB calculation for a horizontal movement. The purple area is the positive OB, the sum of both green areas is the negative OB, and the measure is calculated as the difference between positive to negative divided by their sum.

The VM and the MD measures were selected since they are commonly used measures. OB, PS and AD/MD were defined after initial screening of the velocity and the movement profiles and were based on observed phenomenon. In our experiment using MD was problematic since hemispheres of different radii were tested. Due to physical limitation (CNC machine capabilities in BGU workshop) the radius of the physical hemisphere was 12.5cm while the radius of the virtual hemisphere was 10cm. additional hemisphere manufacturing solutions such as an external production or 3D plastic printing were rejected due to high costs. Thus MD was used only to support VM.

The PS measure was selected since it appeared that many movements were close to straight-line projections in the frontal plane. A plot of all the horizontal and vertical movements (for one subject)

from the physical environment is shown in Figure 11. From this observation the PS measure was set to quantify the distance from the straight-line in both environments. However although the PS measure can describe the similarity to straight line, it doesn't describe whether there was a tendency for a certain direction during the movement. The OB measure was set to answer this question. The AD/MD measure was defined in light of previous finding which showed that in UD movements (pointing and drawing movements) the AD/MD was greater than in DU movements. The OB, PS and AD/MD are normalized measures without units. They range between -1 to 1.

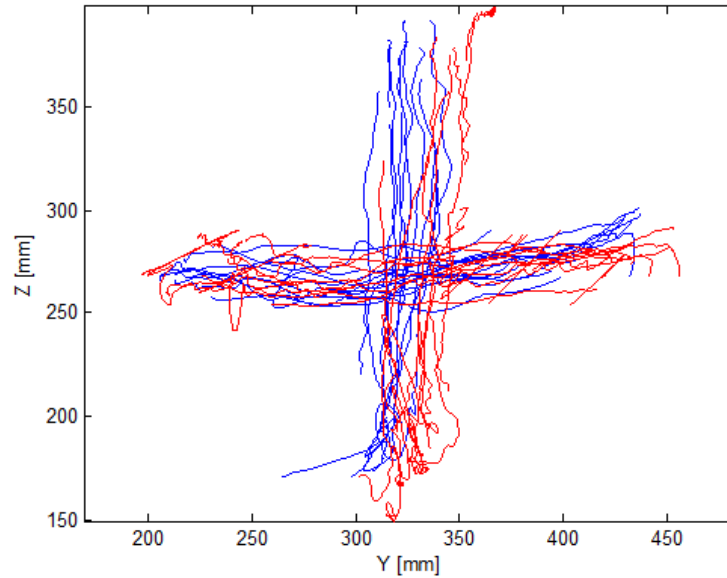


Figure 11: Plotting all the movements for subjects 1 (Blue) and 2 (Red).

3.3 Virtual environment -Subjects and experimental procedure

A constrained motion on a virtual hemispherical surface database was collected from 11 healthy subjects (age 23-56, average 34.1), each subject executed 300 movements and a total of 3300 movements were recorded. The experimental task makes use of a parallel 3D haptic device (Delta.3, Force dimensions) to simulate movements in contact with a hemispherical surface. Subjects grasped the endpoint of the haptic device and were asked to make rapid movements between targets by sliding along the interior of the hemisphere. The hemispherical surface was positioned upright in the subject's frontal plane and was displayed on a computer screen situated approximately 2m from the subject (Figure 12, Figure 13).

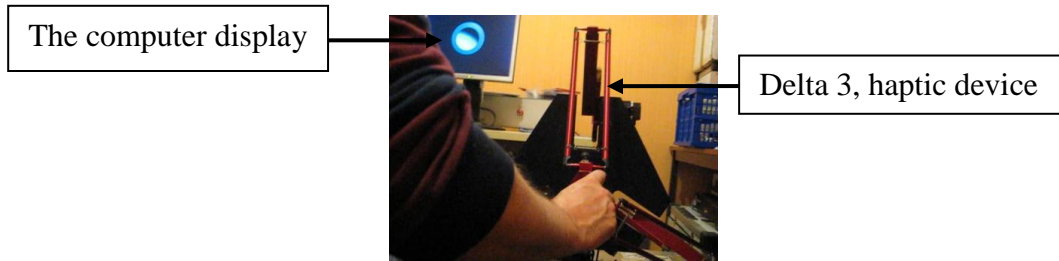


Figure 12: The position of the virtual system compared to the subject.



Figure 13: A – The Delta 3 Haptic device. B – The computer screen.

The experiment was comprised of three sets of 100 movements each. The sets had different spherical radii and distances from the spherical center as detailed in and in Table 2. Each set included movements between 5 targets: center, left, right, up and down. All movement possibilities between these targets were recorded with at least three repetitions of each movement. Each movement started at the position of the previous target. Each set were organized in 5 runs where each run included 20 movements.

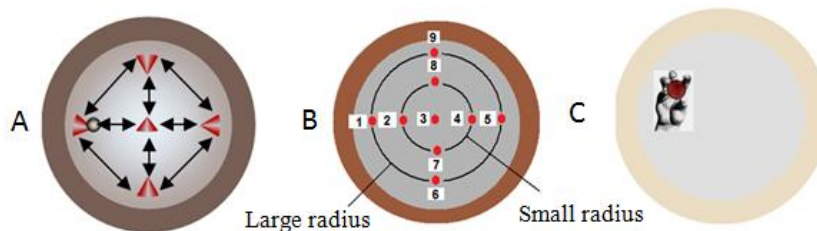


Figure 14: Illustration of movements on the hemisphere.

A – All the movement possibilities. **B** –Targets positions. **C** –The hand space.

Table 2: Experiment sets.

Set	Sphere radii	Distance from center	Points (Figure 14)
LDLR	Large sphere (12.5 cm)	$2/3 * \text{Radius}$	3, 1, 9, 5, 6
LDSR	Large sphere (12.5 cm)	$2/3 * \text{Radius}$	3, 2, 8, 4, 7
SDLR	Small sphere (6.25 cm)	$1/3 * \text{Radius}$	3, 1, 9, 5, 6

Each single movement was executed as follows (Figure 15): The position of the subject on the surface was indicated using a gray conic marker. The conic marker was used to prevent the subject from trying to achieve excessive positioning accuracy. A green conic marker appeared on the screen for 1 second signalling the initial position and the subject moved his hand aligning the position marker with the target marker. Following this a red conic marker appeared on the screen at the target location. The marker was visible for 1 second. The subject started motion towards the target only after the marker disappeared once again to avoid excessive positioning accuracy. After 3 seconds the green marker appeared once more. Every motion started at the target position of the previous motion. The motions in each run were executed consecutively. The subjects were allowed to rest at will between runs. Before recording started, the subjects were allowed to practice until they felt comfortable with the experimental protocol. During the practice trials the subjects received oral feedback regarding the force they were applying (excessive, sufficient etc.). The tip position of the haptic device was recorded along with the force it exerted. This force is equal to the radial force exerted by the users.

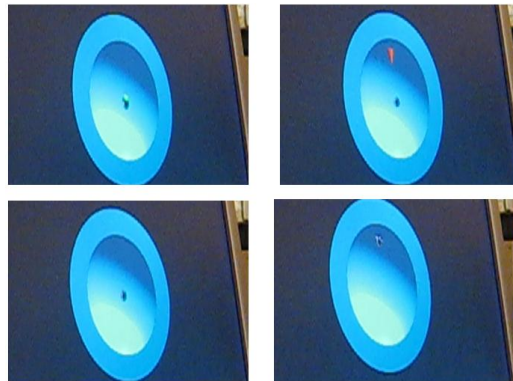


Figure 15: Movement from the center target to the upward target in the virtual environment. Starting from top left in a clockwise manner: Green marker signs the initial position; Red marker signs the target position; Red marker disappears and signs movement onset; Subject reaches target position.

3.4 Physical environment - Subjects and experimental procedure

A constrained motion on a physical hemispherical surface database was collected from 4 healthy subjects aged 24 to 48. Each subject executed 72 movements. The task was modeled after the virtual movement task. Due to fatigue issues in the physical experiment and according to the analysis of the virtual task only four targets were retained in the physical experiment (left, right, up and down) and the central target was excluded (Figure 16). Additionally only a large hemisphere was used and the targets were positioned at a distance of $\frac{2}{3}$ of sphere radius from the center. Each set was organized into 6 runs with 4 movements each (24) and included all the movement possibilities twice. Each of the 6 by 4 set, was repeated three times (thus 72 movements). Every motion started at the target position of the previous motion. Every 4 motions in each run were executed consecutively. The targets were marked on the hemispherical surface. They were concealed from view when the subject was close to the target point thus excessive position accuracy was prevented (similar as possible to the virtual experiment). During the movements the subjects maintained contact with the surface. After reaching the target they would stop and wait for an oral directive to which target to continue. The motions in each run were executed consecutively. The subjects were allowed to rest at will between runs. Before recording started, the subjects were allowed to practice until they felt comfortable with the experimental protocol.

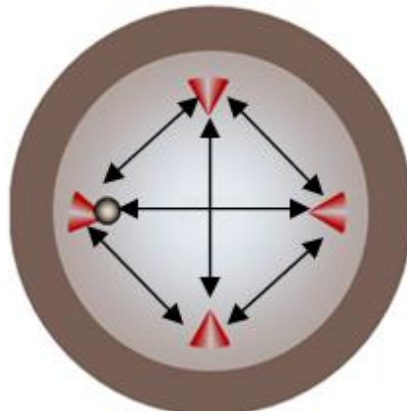


Figure 16: Target positions and movements on the physical sphere

The motions were recorded using a Codamotion system (Charnwoods Dynamics LTD., UK) where two overhead CX1 units were used (Figure 17). Force was recorded using a 6DOF mini40 force sensor (ATI Industrial Automation).



Figure 17: Codamotion CX1 overhead unit.

The hemisphere was made out of Aluminum alloy (Figure 18). Only one side on the hemisphere was machined to reduce manufacturing costs. The radius of the sphere was set to 10 cm due to limitations of the available CNC machines in the University. The possibilities of production by external CNC manufacturers or making the hemispheres from plastic by 3D printing were rejected due to high costs.

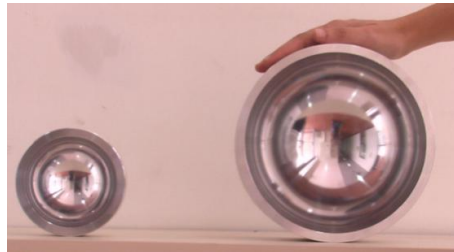


Figure 18: Two physical hemispheres. On the left the large unit

Four Codamotion markers were placed on the hemisphere's rim (up, down, left right) and their position was recorded at the beginning of each experiment to facilitate encoding the motion to the actual surface. The hemisphere was attached to the table using dual stick tape. A handle apparatus (Figure 19) was designed to allow smooth motion together with comfortable grip similar to that of the Detla 3D system. The force sensor was fitted between the tip and the handle in the apparatus. The apparatus was produced using 3D printing. A semi-flexible layer was added between the handle and the force sensor to absorb high pressures and protect the force sensor. A wooden plate was attached to the handle for placing Codamotion markers. This was required as the tip was not visible to the camera system throughout the motion. Three markers were placed on the plate to facilitate full 3D reconstruction. The position of the tip relative to these markers was established by recording the plate position in conjunction with an additional marker placed next to the tip.

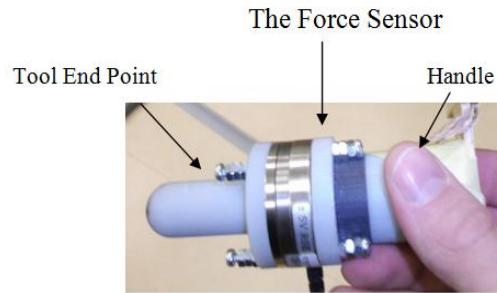


Figure 19: The handle apparatus.



Figure 20: Experimental setup, physical environment. The subject is holding the handle apparatus where only the wooden plate is visible. The Codamotion markers power supplies are placed on a restraining device on the subject's wrist.

3.5 Pre-processing

In both experiments the data was recorded using different position and force sensors. Each movement was recorded into a TXT file, and was uploaded automatically to MATLAB, for the pre-processing stage. In total there were 3300 TXT files in the virtual environment, and 288 TXT in the physical environment. Due to lack of space only an example of the data acquisition appears in Appendix I. Movement analysis required a pre-processing stage. The purpose of this stage is to reduce artifacts and noise-related issues. The pre-processing stage in our research can be divided into four stages:

- 1) Reconstruction of the tip (if needed).
- 2) Filtering
- 3) Segmentation
- 4) Outliers removal

Stage one was only needed in the physical environment due to the use of the physical pointing apparatus. The rest of the stages were done for both environments.

1. Reconstruction of the tip - In the physical environment once the tip (the handle apparatus end effector) entered the hemisphere it was not visible to the Codamotion system which was placed above the rim, Figure 20. To solve this problem, three markers were placed on the back of the tip on a wooden plate, Figure 21. The markers were visible throughout the motion cycle. The tip

position was reconstructed using the markers. Since the wooden plate and the tip were attached rigidly the rotation and the translation of the three markers were identical to those of and the tip. A full formulation of the reconstruction is given in Appendix A. In addition the appendix details the experiments done in order to validate that the reconstruction. The MATLAB code for the reconstruction appears in Appendix O.

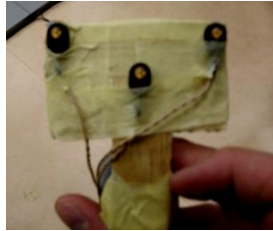


Figure 21: Wooden plate with three markers on back of the tip.

2. **Filtering** – The original data was filtered using a Wiener filter. The Wiener Filter is a filter based on Fourier iteration, proposed by Norbert Wiener in 1949 for reducing the amount of noise in a signal by comparing it to an estimation of desired noiseless signal (Petrou and Bosdogianni, 1999). The Wiener filter was programmed in MATLAB and was chosen due to its known high accuracy while assuming white noise. The frequency of the filter was selected using trial and error.

3. **Segmentation** - Many movements had artifacts at the end of the movement and thus segmentation was needed. For the segmentation pre-process procedures a MATLAB application was developed. A demonstration of the process is shown in Figure 22 A which includes the velocity (Blue) and force (Cyan) profiles and Figure 22 includes the corresponding position profile. Each movement was segmented manually, by moving the pink and the green lines to the corresponding 5% of the movement. In several movements the recording included more than one movement, in such cases, one of the movements was not relevant. That might have happened due to the subject moving his hand after completing the task or before. Thus the corresponding movement was re-plotted in order to validate that the segmentation was correct. In addition in order to save time a simple algorithm for placing the pink and green lines was developed. First the crossing point between the 20% of the velocity amplitude peak and the movement was determined. Then the first two points (one for the start and one for the end) which were 5% of the velocity peak were singled as the starting and end point. This procedure was done automatically and reduced significantly the segmentation time. Movements that included errors such as an early start, zero force and movements at the wrong direction were marked and deleted.

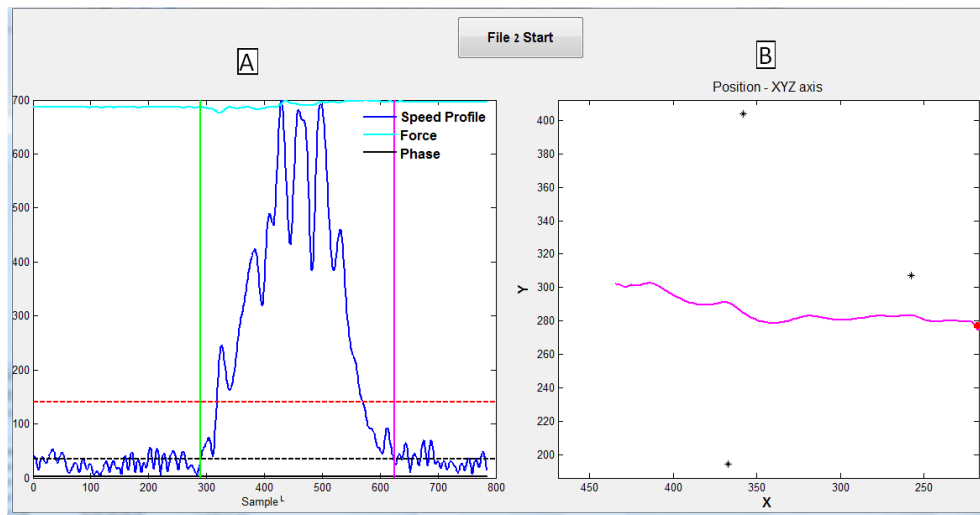


Figure 22: Velocity, force and position profiles of a RL movement with segmentation.

A – Velocity and force profiles. The green and pink lines present the beginning and the end of the movement respectively. The dashed black and red lines are the 5% and 20% of the maximum peak in the velocity profile respectively. **B** – Shows the movement after segmentation. The movement shown is right-to-left. The red point is the starting point.

Figure 23 shows the appearance of a movement without segmentation. Since that movement was not segmented thus a small artifact remained at the end of the movement (marked in red). That artifact was a movement done by the subject after he reached the final target. That movement was not part of the research and thus it was segmented.

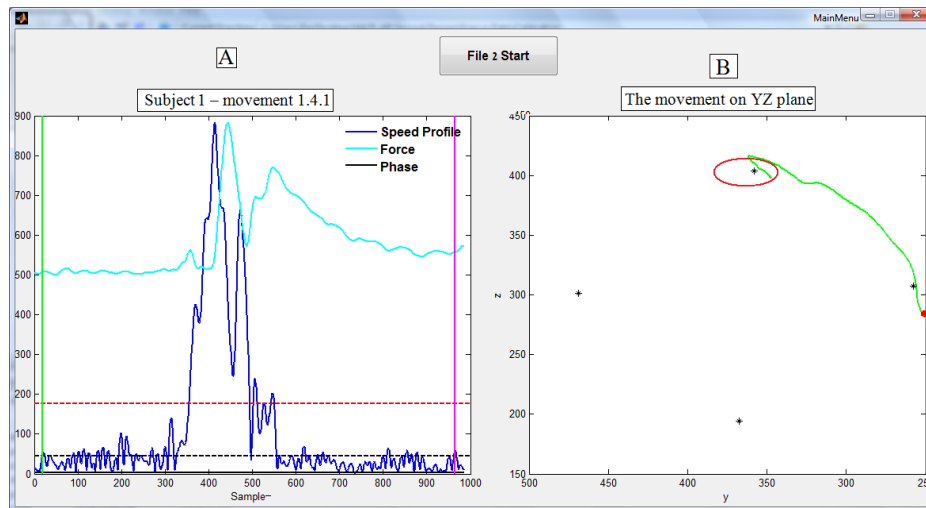


Figure 23: Velocity, force and position profiles of a RU movement without segmentation.

A – Velocity and force profiles. The green and pink lines present the beginning and the end of the movement respectively. The dashed black and red lines are the 5% and 20% of the maximum peak in the velocity profile respectively. **B** – Shows the movement without segmentation. The movement shown is an oblique right-to-up. The red point is the starting point. The red circle is the part that was not segmented.

4. **Outliers removal** - Three types of outliers were defined:

- **Early start** - In the physical experiments each subject had to wait for an oral command while in the virtual experiment the commands were given visually (conic marker which changed colors between stages) in order to start the movement. Several movements started before the oral command and thus were not analyzed further.
- **Zero force** – The task was to maintain contact with the surface. Due to the haptic illusion in the virtual environment we defined that when the measured force was zero than that meant that there was no contact with the surface and the movement was disqualified.
- **Other** – Movements that were in the wrong direction or the subject didn't stop at the target point and thus the velocity profile could not be segmented properly.

The following analysis refers to the distribution of the outliers in both environments and includes only horizontal and vertical movements (UD, DU, LR, and RL). The total number of movements in the virtual environment was 219 (Table 3). 84.02% of the movements in the virtual environment were considered for analysing the measures (no outliers), 13.24% had zero force, 2.29% started too early and 0.46% had other problems and thus were disregarded. In the physical environment 96 movements were recorded (Table 3). 91.67% were considered in the analysis, 8.33% had other outliers and thus were disregarded.

Table 3: Outlier distribution of Horizontal and Vertical movements.

Outlier Type	Virtual Environment		Physical Environment	
	#movements	Percentage	#movements	Percentage
No Outlier	184	84.02%	88	91.67%
Early start	5	2.29%	0	0%
Zero Force	29	13.24%	0	0%
Other	1	0.46%	8	8.33%
Total	219	100.00%	96	100%

Since most of the analysis concentrated on straight lines, movements with a very high radius (meaning 15% of the bowl radius) were also disregarded. Following this step, in the virtual environment 146 of the 184 movements remained, and in the physical environment 77 of 88 movements remained (Table 4). The distribution between the different direction types was similar and around 55 for each direction type. The total distribution analysis of all the movements which were measured in both environments (including movements such as LU, UL...) appears in Appendix K.

Table 4: The number of movements in each direction for every environment

		Physical Environment	Virtual Environment	Total
Movement Direction	DU	21	38	59
	LR	18	33	51
	RL	16	37	53
	UD	22	38	60
	Total	77	146	223

3.6 Results and statistical analysis

The statistical analysis was performed using SPSS version 16. In the following section each performance measure is analyzed separately, starting with a three-way ANOVA. In the three-way ANOVA the fixed factors were movement direction, subject ID, and environment. Table 5 summarizes the three-way ANOVA results..

When needed, a post hoc analysis was used to find patterns in the movement direction interactions, for the post hoc analysis the Fisher's Least Significant Difference (LSD) was selected. The LSD test is equivalent to multiple individual student t-tests between all pairs of groups. The LSD test makes no attempt to adjust the observed significance level for multiple comparisons.

Table 5: Three-way ANOVA results.

Measure	VM	OB	PS	AD _{total} /MD	AD _{rel} /MD	MD
	P-value	P-value	P-value	P-value	P-value	P-value
Subject ID	.155	.007	.172	.262	.126	.000
Environment	.000	.240	.000	.000	.949	.000
Movement Direction	.000	.000	.082	.000	.000	.021
(Subject ID)*(Environment)	.000	.019	.716	.254	.585	.000
(Subject ID)*(Movement Direction)	.997	.035	.979	.364	.950	.003
(Environment)*(Movement Direction)	.000	.264	.000	.000	.580	.000
(Subject ID)*(Environment)*(Movement Direction)	.449	.587	.493	.681	.426	.196

3.6.1 Velocity Mean (VM)

The three-way ANOVA for the VM measure shows that the subject ID was not significant (P-value=0.155, Table 5), however the results for the MD measure show that the subject ID is significant and since both measures imply similar conclusions. A short period of movement time implies that the movement speed is high. Thus the MD measure would be used to strengthen the VM measure.

As subject ID was found not significant a two-way ANOVA was performed using the environment and the direction as the factors was performed (Table 6). The two-way ANOVA shows that the movement direction, the environment and their interaction are all significant (P-value=.000). Therefore two one-way ANOVA's with movement direction as factor were performed for each environment. Movement direction in both environments was significant (physical environment: P-value=.000, virtual environment: P-value=.023).

Table 6: VM two-way ANOVA.

VM	P-value.
Movement Direction	.000
Environment	.000
(Movement Direction) * (Environment)	.000

The LSD post-hoc for testing the movement direction in each of the one-way ANOVA shows that indeed there is significance in the movement direction and that this significance is different in every environment, Table 7.

Table 7: LSD post hoc analysis for movement direction of the one-way ANOVA, VM measure.

Environment	DU (P-value)			LR (P-value)			RL (P-value)			UD (P-value)		
	LR	RL	UD	DU	RL	UD	DU	LR	UD	DU	LR	RL
Physical	.004	.954	.002	.004	.008	.000	.954	.008	.003	.002	.000	.003
Virtual	.176	.497	.079	.176	.047	.003	.496	.047	.284	.079	.003	.284

The error bar for the average VM (Figure 24) shows that the movements in the physical environment (regardless of the direction) were significantly faster than movements in the virtual environment. Figure 25 Shows MD behaves similarly to VM.

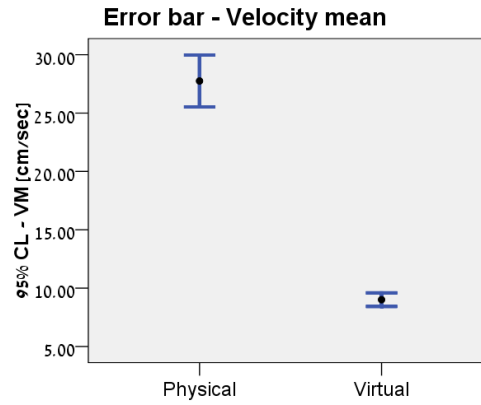


Figure 24: Error bar with 95% confidence level for the average VM measure.

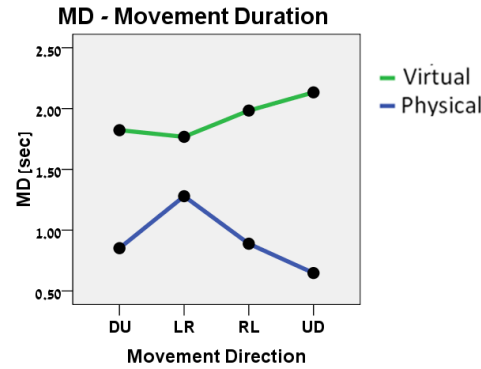


Figure 25: MD plots for both environments.

3.6.2 Statistical analysis - AD_{total}/MD

The three-way ANOVA for the AD_{total}/MD measure shows that the subject ID was not significant (Table 5, P-value=.262) and thus a two-way ANOVA with the movement direction and the environment as factors was performed (Table 8). The Two-way ANOVA for the AD_{total}/MD measure shows that the movement direction, the environment and the interaction were significant (P-value=0.000). Therefore two one-way ANOVA's with movement direction as factor were performed for each environment. Movement direction in both environments was significant (physical environment: P-value=.000, virtual environment: P-value=.000).

Table 8: AD_{total}/MD two-way ANOVA

	P-value.
Movement Direction	.000
Environment	.000
(Movement Direction) * (Environment)	.000

The LSD post hoc analysis for the movement direction in each the one-way ANOVA's is shown in Table 9. In the **physical** environment there is a significant difference between UD and DU movements (P-value=.000), but there is no significant difference between LR and RL movements (P-value=.413), Table 9. However, in the **virtual** environment that finding does not hold. There is no significant difference between DU and UD movements (P-value=.190). The only main difference is between LR and the rest of the movement directions, Table 9.

Table 9: LSD post hoc analysis for movement direction of the one-way ANOVA, AD_{total}/MD measure.

Environment	DU (P-value)			LR (P-value)			RL (P-value)			UD (P-value)		
	LR	RL	UD	DU	RL	UD	DU	LR	UD	DU	LR	RL
Physical	.001	.000	.000	.001	.413	.000	.000	.413	.000	.000	.000	.000
Virtual	.000	.411	.190	.000	.000	.001	.411	.000	.032	.190	.001	.032

Figure 26 shows the average result combined with an error bar of a 95% confidence level of the AD_{total}/MD measure for each direction. It is very clear from Figure 26 that there is a-symmetry between UD and DU in the physical environment which fits to previous research of the measure in drawing and pointing movements. This a-symmetry is not shown in the virtual environment.

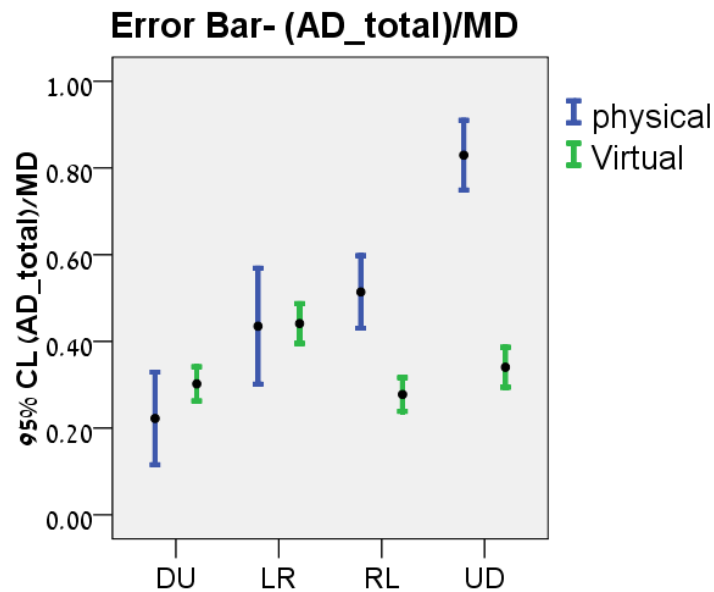


Figure 26: Error bar with 95% confidence level for the average (AD_{total}/MD) measure. AD_{total}/MD is a normalized measure without units. It ranges from -1 to 1.

3.6.3 Statistical analysis - AD_{rel}/MD

This measure was defined according to the results of the AD_{total}/MD measure. The results for the physical environment showed that the AD_{total} /MD measure was as expected (from the literature review) greater for UD movements then for DU movements. However the results in the virtual environment were surprising and thus we decided to test the AD/MD measure slightly differently.

The three-way ANOVA for the AD_{rel}/MD measure showed that the subject ID was not significant (Table 5, P-value=.126) as in the AD_{total}/MD measure. Thus repeating the analysis of the two-way ANOVA's using movement direction and environment as the factors was done. The movement direction was significant (P-value=.000), but the environment (P-value=.986), and the interaction were not significant, Table 10. Therefore two one-way ANOVA's with movement direction as factor were performed for each environment. Movement direction in both environments was significant (physical environment: P-value=.000, virtual environment: P-value=.000).

Table 10: Between subjects test, PS measure for both environments.

	P-value.
Movement Direction	.000
Environment	.986
(Movement Direction) * (Environment)	.475

Since the environment is not significant, an LSD post analysis was performed for movement direction of the two-way ANOVA. The results show that horizontal movements (LR and RL) were not significantly different from each other, but vertical movements (UD and DU) were significantly different from the rest, Table 11. Thus an error bar with the 95% confidence level was done for both environments separately (in order to see that the results were similar between both environments), Figure 27. In addition it can be seen from the figure that there is a-symmetry between UD and DU movements, and symmetry between LR and RL movements. This result fits the previous research which was done on the AD/MD measure.

Table 11: LSD post hoc analysis for movement direction of the two-way ANOVA, AD_{rel}/MD measure

	DU (P-value)			LR (P-value)			RL (P-value)			UD (P-value)		
	LR	RL	UD	DU	RL	UD	DU	LR	UD	DU	LR	RL
P-value	.002	.000	.000	.002	.124	.000	.000	.124	.000	.000	.000	.000

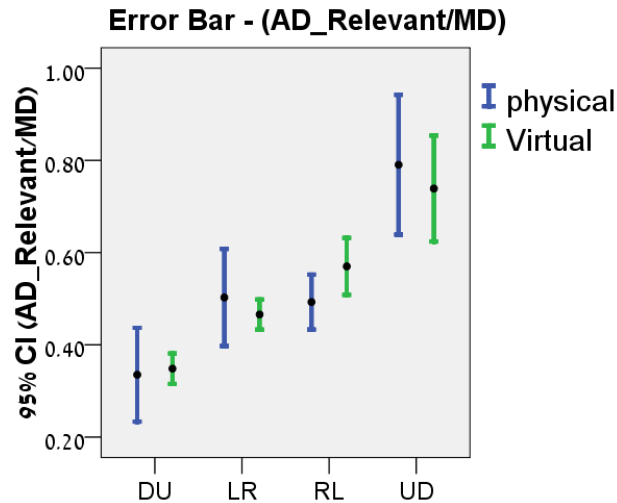


Figure 27: Error bar with 95% confidence level for the average (AD_{rel}/MD) measure. AD_{rel}/MD is a normalized measure without units. It ranges from -1 to 1.

3.6.4 Statistical analysis - Path Straightness (PS)

The three-way ANOVA for the PS measure shows that the subject ID was not significant (P-value=0.172, Table 5). Thus another two-way ANOVA analysis using the movement direction and the environment as the factors was performed. The results show that the movement direction, the environment and the interaction were all significant (P-value<0.05, Table 12). Thus two one ANOVA's with movement direction as the factor were performed and the results showed that movement direction was significant in both environments (physical environment: P-value=.001, virtual environment: P-value=.000).

Table 12: Between subjects test, PS measure for both environments.

	P-value.
Movement Direction	.002
Environment	.000
(Movement Direction) * (Environment)	.000

An LSD Post hoc analysis for movement direction of the one-way ANOVA for each environment shows a significant effect in RL and LR movements when compared to UD and DU movements (Table 13). The average PS results shows that in the virtual environment all the movements were closer to the straight-line projection (PS=1) then in the physical environment. In addition there is an opposite symmetry behaviour between the environments in the vertical and horizontal movements, Figure 28.

Table 13: LSD post hoc analysis for movement direction of the one-way ANOVA, PS measure.

Environment	DU (P-value)			LR (P-value)			RL (P-value)			UD (P-value)		
	LR	RL	UD	DU	RL	UD	DU	LR	UD	DU	LR	RL
Physical	.004	.001	.526	.004	.622	.017	.001	.622	.001	.526	.017	.005
Virtual	.000	.000	.830	.000	.768	.000	.000	.768	.000	.830	.000	.000

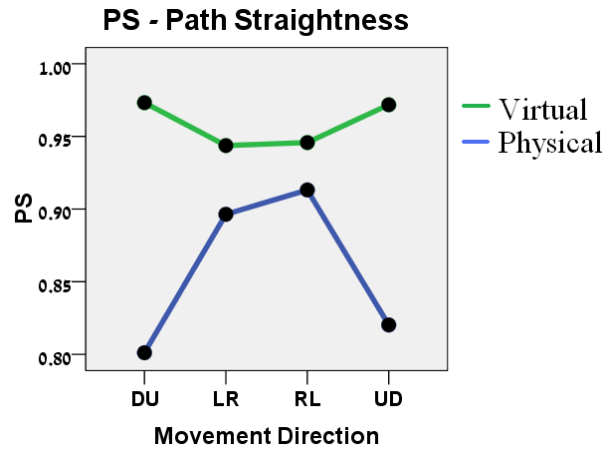


Figure 28: PS plots for both environments.

PS is a normalized measure without units. It ranges from -1 to 1.

This interesting result is emphasized in Figure 29 which shows the error bar plot of the average PS measure for the vertical Vs horizontal movements in both environments. The main effect which is shown is that in general movements in the virtual environment were straighter than movements in the physical environment.

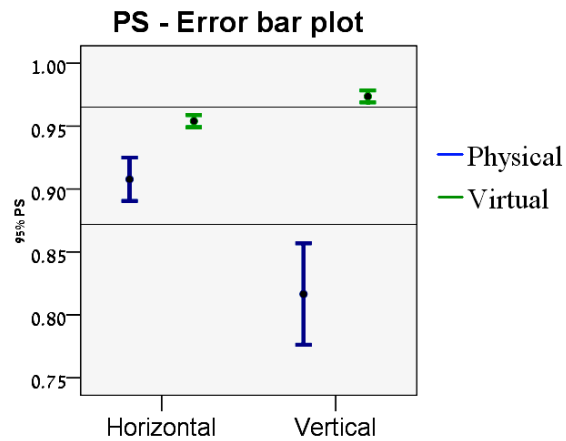


Figure 29: Error bar for horizontal and vertical movements, PS measure. No units, ranges from -1 to 1.

3.6.5 Statistical analysis - Orthogonal bias (OB)

The three-way ANOVA for the OB measure (Table 5) showed that there was significant effect for the subject ID (P-value=.007), the movement direction (P-value=.000) and that there was an interaction between both (P-value=.019). There for movement direction was separated into two smaller groups (vertical and horizontal). Three-way ANOVA for only vertical movements (UD, DU) showed that there is no significance for the subject ID (Table 14) which means the subject dependency appeared in the horizontal movements. Thus indeed three-way ANOVA for the horizontal movements shows that there is a significant effect for the subject ID (P-value=0.000).

Table 14: Three-way ANOVA for vertical and horizontal directions separately.

	Vertical	Horizontal
	P-Value	P-Value
Subject ID	.307	.000
Environment	.008	.924
Movement Direction	.781	.008
(Subject ID)*(Environment)	.045	.006
(Subject ID)*(Movement Direction)	.182	.012
(Environment)*(Movement Direction)	.651	.094
(Subject ID)*(Environment)*(Movement Direction)	.681	.094

An average plot of the OB measure for the horizontal movements in the virtual environment shows that subject number 5 in the LR movements and subject number 7 in the RL movements were different than the rest and moved above a straight-line projection on a the planar surface of the hemisphere instead of beneath it, Figure 30. An LSD post hoc analysis for movement direction in the three-way ANOVA for only LR movements shows that only subject number 5 was significantly different (Table 15), the full table appears in Appendix D. The same LSD post hoc for the RL movements showed that subject 7 is significantly different than the rest, Table 16.

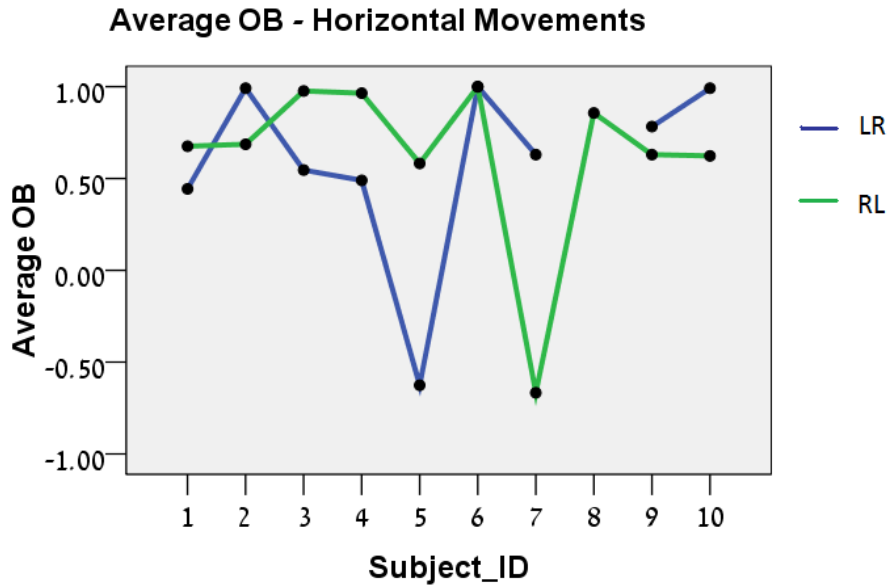


Figure 30: Average OB measure Vs the subject ID for horizontal movements in the virtual environment. OB is a normalized measure without units. It ranges from -1 to 1.

Table 15: Partial LSD post hoc analysis for the OB measure, LR movement.

Source	Subject 1							
Subject	2	3	4	5	6	7	9	10
P-value	.111	.728	.890	.002	.106	.578	.255	.111
Source	Subject 5							
Subject	1	2	3	4	6	7	9	10
P-value	.002	.000	.000	.003	.000	.001	.000	.000

Table 16: Partial post hoc analysis for the OB measure, RL movement.

Source	Subject 1								
Subject	2	3	4	5	6	7	8	9	10
P-value	.962	.245	.264	.673	.174	.000	.442	.857	.858
Source	Subject 7								
Subject	1	2	3	4	5	6	8	9	10
P-value	.000	.000	.000	.000	.000	.000	.000	.000	.000

Although there is significance for the subject ID, this result appeared only in several special cases. The average OB measure for both environments shows that there was a tendency downwards in horizontal movements (positive OB is with the direction of the gravity), while in the vertical movements there was no such tendency, Figure 31, Figure 32. This result is an average result and it was shown that one subject out of ten (in two different directions) didn't fit to this conclusion.

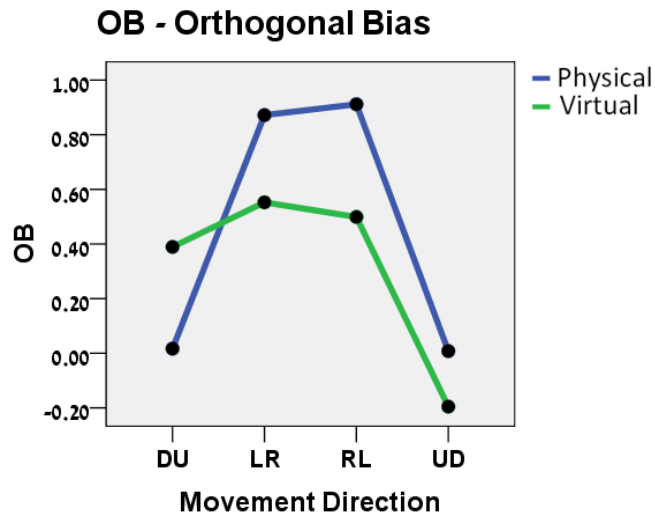


Figure 31: OB plots for both environments. No units, ranges from -1 to 1.

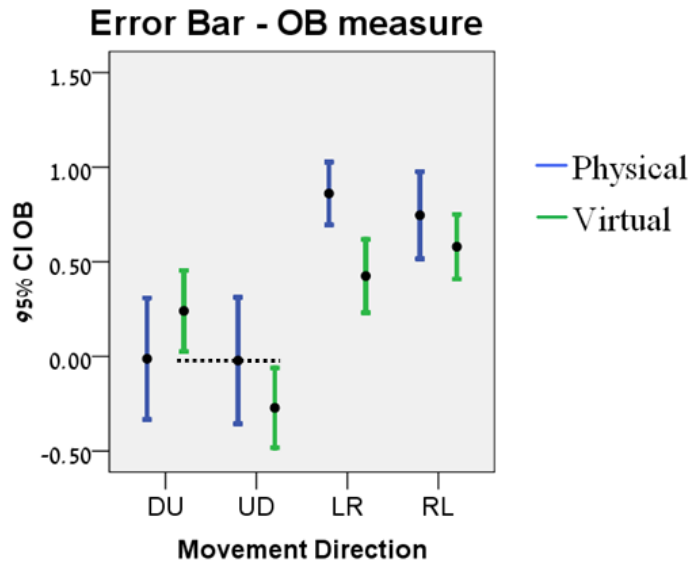


Figure 32: Error bar for horizontal and vertical movements, OB measure. Ranges [-1:1].

3.7 Discussion

Environment findings – All the measures showed that there were differences between the environments. The first difference that was found was that in the virtual environment 13.24% of the movements included zero force, while in the physical environment there were no such movements.

The PS measure shows that the vertical movements in the physical environment were closer to straight lines than the horizontal movements. This difference could be attributed to the fact that the movements in the virtual environment were significantly slower, and thus more accurate. In addition the movements in the virtual environment were closer to straight-line projections. Together with the OB measure this difference could be explained by the lack of DOF endpoint orientation in the virtual environment. Another explanation could be the torque which was created by the handle apparatus. In the virtual environment the handle was part of the system and thus didn't create any torque, while in the physical environment the handle apparatus created a larger torque. During the physical experiment fatigue issues appeared and might be the cause for the stronger tendency downwards in vertical movements in the physical environment. In addition another important difference was that in the virtual environment the handle apparatus was fixed in a way that the subject could not rotate the handle end point, while in the physical environment the end point was free to rotate. This actually reduced the DOF in the virtual environment and could be the cause for the straighter lines in the virtual environment. The last difference in the results between the environments was that movements in the physical environment were significantly faster than in the virtual environment. This difference could be attributed to the difference in the sensation feedback. In the virtual environment the feedback was done by a computer screen, and the hand was located in para-frontal plane. Such movement is not natural and thus it is possible that the subjects were less confident in the movement, which corresponded in a slower movement in order to reach the final target, while in the physical environment the movement was natural and similar to an everyday task.

Movement Direction findings - our findings, regarding the movement directions, are in line with previous research which found similar asymmetries during whole body motion, (Papaxanthis et al., 2003), during pointing movements (Papaxanthis et al., 1998) and in gravity fields (Gentili et al., 2007). The acceleration peak (from the AD/MD measure) appears earlier in DU movements compared to UD movements when looking on the main movement axis. Since the results were found in both environments and in different speeds, those results suggest that gravity is taken into account in the motor planning and execution of movements on hemispherical surface. In other words, although there

are great obvious kinematic differences which could explain the results for each environment separately, the different movement speeds and previous research which showed that those results also appeared in pointing different movement speeds (Papaxanthis et al., 2003) and without gravitation (Papaxanthis et al. 2005) may indicate that the gravity is taken into account by the CNS during motion planning.

The OB measure showed another effect of the gravity. Horizontal movements had a tendency downwards while such tendency was not found in vertical movements. In the physical environment this result was significant, however in the virtual environment it was not significant, due to two subjects out of twenty (one out ten in a LR movement and one out ten in a RL movement, not the same subject) which tended to move above the straight-line projection. Although the difference between the environments could be explained by the larger torque applied on the hand in the physical environment, it does not explain the two subjects which acted differently. We have no explanation for this at this point and it requires testing with a larger set of subjects.

According to all the measures it seems that there were strong differences between the environments. However, most of the differences have a strong physical explanation and thus those differences actually strengthen the validation of the virtual environment.

Chapter Four: Robotic constrained motion

The robotic experiment was designed in order to validate findings of the human experiments. At this stage the main questions that this experiment tried to answer were: Is it possible to use hybrid control for constrained motion on a hemisphere? Could we support hypothesis regarding the use of such control policy for human motion control? Could we find evidence for gravity effects on robotic motion?

This chapter begins with detailing the performance measures followed by section 4.2 which presents the notion of *Position Based Hybrid Control (PB-HC)* and discusses the advantages of the hybrid control Vs impedance control for moving on a constrained surface. Section 4.3 gives the experimental description followed by a description of the three different algorithms used for applying the PB-HC. Finally the results and the discussion are presented in sections 4.4 and 4.5.

4.1 Performance Measures for the robotic experiment

The measures were defined in light of two important issues: what can be learned in comparison to human movement and which of the algorithms is more suited for the constrained motion task on the other hand. The comparison to human motion is only partial since the robotic control loop is too slow in the used manipulator. This caused a very noisy velocity profiles (Figure 33) and prevented quantitative comparison to human profiles. Regardless, qualitative comparison of PS, OB, and MD is possible.

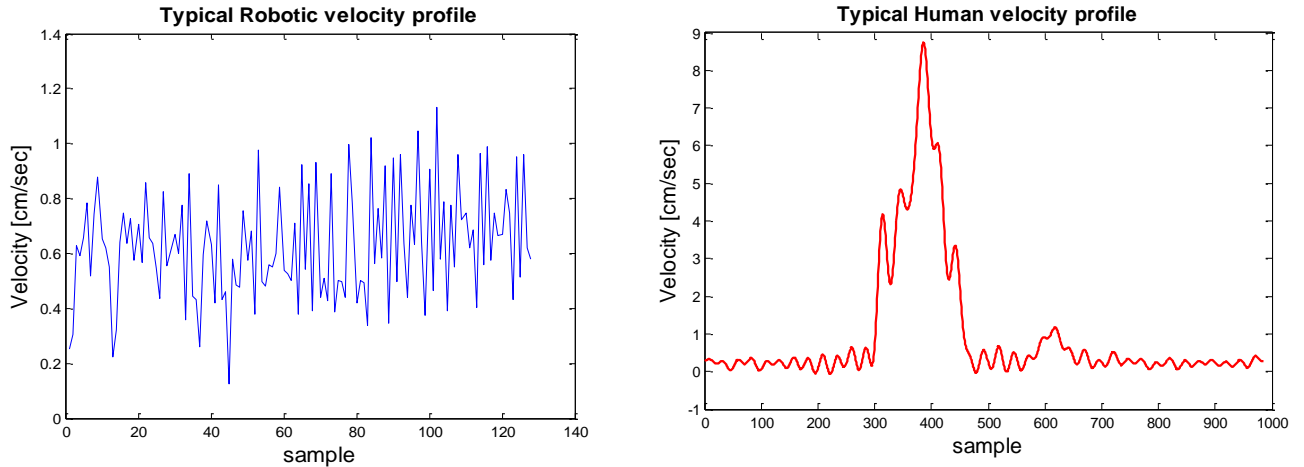


Figure 33: Velocity profiles for a robotic and a human movement.

Two additional measures were defined for comparing the performance of the different control algorithms:

1) **Number of detachments from the hemisphere (ND)** – ND was calculated using force measurements. First the force was measured without contact with the surface (F_{zero}). This force is not zero and changes every run since the sensor is sensitive to the weight of the apparatus, its orientation and the way it is attached to the gripper. The profile measurements onset is defined when contact with the surface is made and the measured force is greater than F_{zero} . Each time the measured force crosses F_{zero} it is counted as a detachment.

2) **Force Range (FR)** – FR calculates the difference between the maximum and the minimum applied force. A smaller FR is preferred.

4.2 Experimental Setup

Robotic constrained motion was implemented using a manipulator with 5 DOF (Scorebot Er 5+) and a force sensor (Mini 45, ATI). The force sensor was placed in the handle apparatus used in the physical environment, Figure 19. The apparatus was rigidly attached to the gripper of the robotic manipulator (Figure 34), using tape for ensuring handle stability. The force sensor and the manipulator were programmed using MATLAB the sample rate of the force was 100Hz.

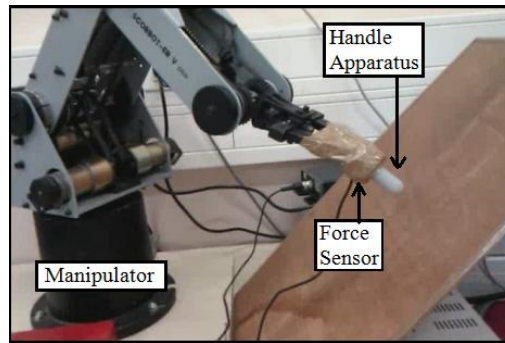


Figure 34: The apparatus attached to the manipulator.

In addition to the previous assumptions it was assumed that the surface shape is unknown, thus the algorithm developed would be suited for different types of environments and no a-priori trajectory planning is involved. The suitability to other environments was validated on a planner tilted surface (Figure 34). Figure 35 shows the axes used throughout the experiment. The force feedback was given in the X axis. The force was translated to a position command according to the implemented algorithm. The movement in Y and Z were controlled according to the target coordinated.

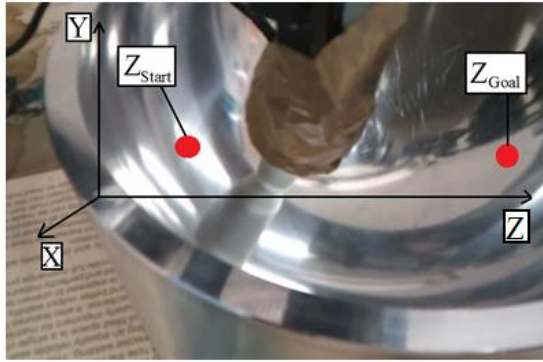


Figure 35: The axis in the experiment.

For each algorithm four types of movements (LR, RL, UD and DU) were tested and compared.

4.3 Position Based Hybrid Control (PB-HC)

Since the Scorebot Er5+, like most industrial robots, has a position based controller, the hybrid control algorithm needs to transform the force command into a position command. Thus a position based hybrid control (PB-HC) scheme was developed (Figure 36). The control scheme has two main loops which are combined into one. In this diagram Y and Z apply the position control and X is used for the force control (force is translated into position command). The axes may change if needed for other purposes as long as they remain orthogonal. The lower block diagram includes an inner loop when q_d refers to the engine desired angle and q_r refers to the engine real angle. q_r is transferred by forward kinematics to Y and Z positions in space, and thus allows the selection of a new target point directed to the final position. In parallel there is the upper loop which uses the force sensor. F_{int} is the interaction force from the sensor. It is compared to the desired force ($F_{Desired}$) and by a transformation function it is transferred to a position command (can be done in many ways such as constant stepping, linear proportion and more). The final step is to transfer the new desired X, Y and Z (X resulted from the force) into motor command by inverse kinematics.

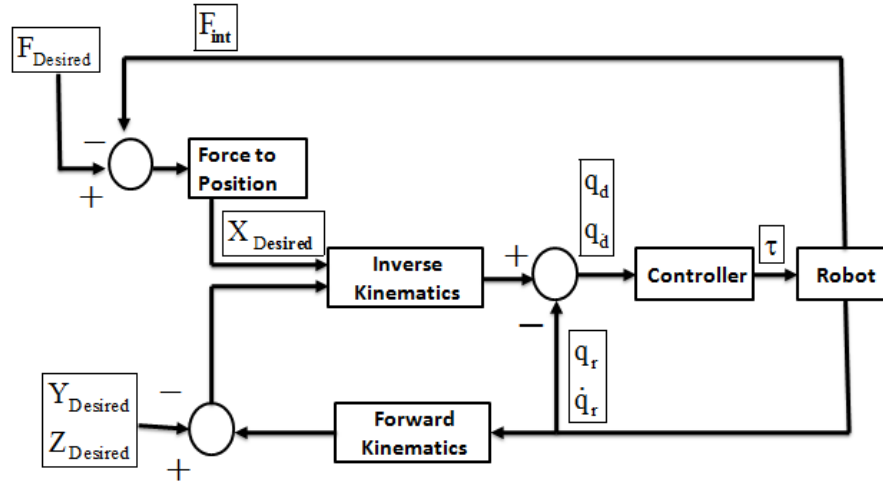


Figure 36: Position Based Hybrid Control, block diagram. q_d and q_r are the engines desired and real positions. F_{int} is the interaction force and $F_{Desired}$ is the wanted force. $X_{Desired}$ and $Y_{Desired}$ are the wanted positions while $Z_{Desired}$ is achieved by transferring force to position.

4.3.1 Hybrid Vs Impedance Control for applying constrained motion

The investigated task was constrained motion on a hemisphere. The task can be divided into two goals:

- 1) Position goal - Reaching from point to point.
- 2) Force goal - Maintaining contact during **the entire** movement.

The force goal should be satisfied at all time. However the trajectory is not predefined, thus the controller isn't required to track a specified path. It is possible to time and the energy of the task, but in our case a compromise between the force and the position was selected and thus every trajectory is acceptable as long as it reaches the final goal and maintains contact during the entire movement.

Under this definition of the task and using the assumption that the force must be maintained during the entire movements, there are few basic issues for comparing the hybrid and the impedance control methods:

Kinematics - Both methods (hybrid and impedance) requires the use of inverse and forward kinematics.

Dynamics - Impedance control requires an accurate dynamic model, while hybrid control can avoid dynamics if the task speed is slow or if the gear ratio is high, (Antonio et al., 2009).

Tracking – Impedance control tracks the impedance that is changing according to the interaction force in time (in position based impedance and due to hardware limitations that is what we have). Thus it doesn't track a specified force or a specified path only the correlation between the two (including the Instantaneous model). Hybrid control can track either one of the position or force control while preferring one limitation on top of the other. In light of the task it is possible to demand a certain force during the entire movement, and still to reach the goal.

From this comparison the main issue which arises is that the impedance control follows the impedance and not the force, the hybrid control method was selected in order to apply the task.

4.4 The Hybrid Algorithms

Three variations of hybrid control were implemented:

- 1) Hybrid control with constant stepping (CS control).
- 2) Proportional stepping (P control).
- 3) Proportional Differential stepping (PD control).

All three algorithms had the same basic structure and the same control diagram as shown in Figure 36, as a result they all share the same basic goals of the hybrid control:

- Position goal – To reach from one point to another, and not to track a specified trajectory.
- Force goal - To maintain a constant force desired force which is larger than zero but would not harm the force sensor $\rightarrow e=f_0 - f_{\text{final}}$.

All three algorithms were programmed with a similar pseudo code which is illustrated in Figure 37. The difference between the algorithms is in the transfer function of the desired force into a position command. Forces that were measured without any contact with the surface were not zero since the sensor was influenced by the weight of the apparatus and the tape which pressured the sensor. Thus in all cases the force which was measured before any contact was made, was referred to as Fzero (although it was not) and the forces which were measured afterwards were defined relatively to Fzero.

• Comment – The Scorebot Er5+ comes with an internal controller which converts voltage into encoder reading and vice versa. An equation was developed to transfer encoder readings into motor angles. That enabled reading the motor angles.

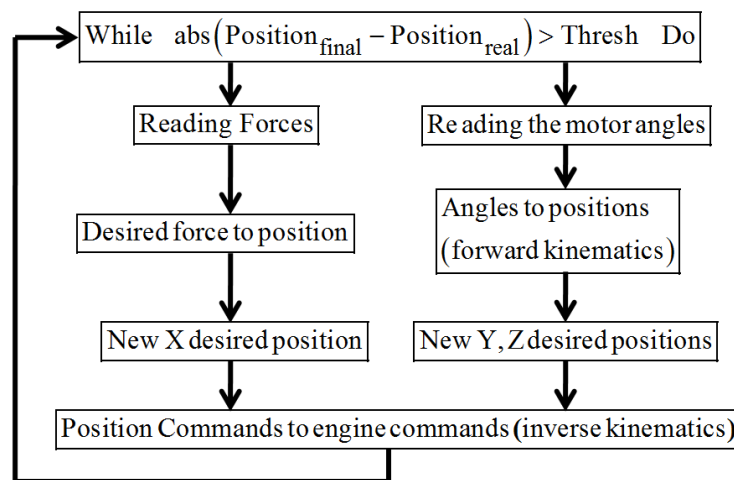


Figure 37: Pseudo code illustration for the applied Hybrid control.

4.4.1 Hybrid control with Constant Stepping (CS)

The basic structure of the first algorithm included constant stepping (CS) and a threshold cut-off. This was in fact bang-bang control applied in a hybrid control scheme. The position command in the X direction was constant until the force crossed the given threshold and at that point it changed from an inside step to an outside step. Table 17 illustrates the algorithm. Once the force is larger than a predefined threshold (defined by experimental trial and error) then the command in X was to update the position in a constant step which was further from the constraint. Y and Z axes were also updated with the same constant step towards the final target. Similar analogy was made for the rest of the states, once the force was smaller thus the position commands were made in given in order to increase the applied forces and in a medium the position commands were given so that the force would remain the same.

Table 17: Hybrid control with CS, commands summary.

Force amplitude	X (Force controlled)	Y (Position Controlled)	Z (Position Controlled)
Large	Constant step away from the constraint.	Constant step towards the final target.	Constant step towards the final target.
Medium	No step.	Constant step towards the final target.	Constant step towards the final target.
Small	Constant step towards the constraint.	No step.	No step.

The step size was determined empirically. When determining the step size a compromise was sought between MD and the ND. Constant steps of 1mm, 1.5mm and 2mm were evaluated.

4.4.2 Hybrid control with Proportional (P) stepping

In this algorithm proportional (P) control rather than bang-bang control was applied (equation 4.1).

$$(4.1) \quad \text{Step} = \frac{\text{Max}(\text{step}) - \text{Min}(\text{Step})}{\text{Max}(F_{\text{measured}} - F_{\text{Zero}})} \cdot (F_{\text{measured}} - F_{\text{Zero}}) + \text{Max}(\text{step})$$

The equation parameters were set empirically and were chosen as follows:

$$\text{Max}(\text{step}) = 3 \quad ; \quad \text{Min}(\text{Step}) = -3$$

This equation allowed the step size to vary between:

$$-3\text{mm} = \frac{6}{15} \cdot (-15) + 3 < \text{Step} < \frac{6}{15} \cdot 0 + 3 = 3\text{mm}$$

Thus most of the steps should be around the same step size as defined in the first two experiments. In order to avoid complications which may be caused due to higher forces (which can result in bigger steps), the steps were limited to their boundaries of $\pm 3\text{mm}$. The position control was set to constant step of 1.5mm . Illustration of the algorithm is shown in Table 18.

Table 18: Hybrid control with P stepping, commands summary.

Force size	X (Force controlled)	Y (Position Controlled)	Z (Position Controlled)
Large	P Stepping	Constant step towards the final target.	Constant step towards the final target.
Medium		Constant step towards the final target.	Constant step towards the final target.
Small		No step.	No step.

4.4.3 Hybrid control with Proportional Differential (PD) Stepping

The P algorithm had overshoot problems when the proportion constant was very big. Adding a differential controller is a classical improvement of the algorithm (a common effect in P control). Thus PD control was used to solve the overshoot issue (equation 4.2).

$$(4.2) \quad \text{Step} = \frac{\text{Max}(\text{step}) - \text{Min}(\text{Step})}{\text{Max}(F_{\text{measured}} - F_{\text{Zero}})} \cdot (F_{\text{measured}} - F_{\text{Zero}}) + \text{Max}(\text{step}) + K_d \cdot \frac{(F_{\text{measured}} - F_{\text{Zero}})}{dt}$$

K_d is the differential constant which was empirically selected. It was very small since dt was around 0.16sec. An illustration of the algorithm is shown in Table 19.

Table 19: Hybrid control with PD stepping, commands summary.

Force size	X (Force controlled)	Y (Position Controlled)	Z (Position Controlled)
Large	PD Step	Constant step towards the final target.	Constant step towards the final target.
Medium		Constant step towards the final target.	Constant step towards the final target.
Small		No step.	No step.

4.5 Results

4.5.1 PS and OB measures

The results in all the movement directions show that the hybrid control (in all three algorithms and different step size) created straight-line path in the frontal plane, (Figure 38). The similarity to straight lines was measured by the PS measure, which in all cases (algorithms and directions) was above 90% (Table 21). In addition, for vertical movements (UD and DU) for all movements PS was above 99%. The PD algorithm was the closest one to a straight-line when taking into account all the directions (especially in the LR and RL movements), Table 21.

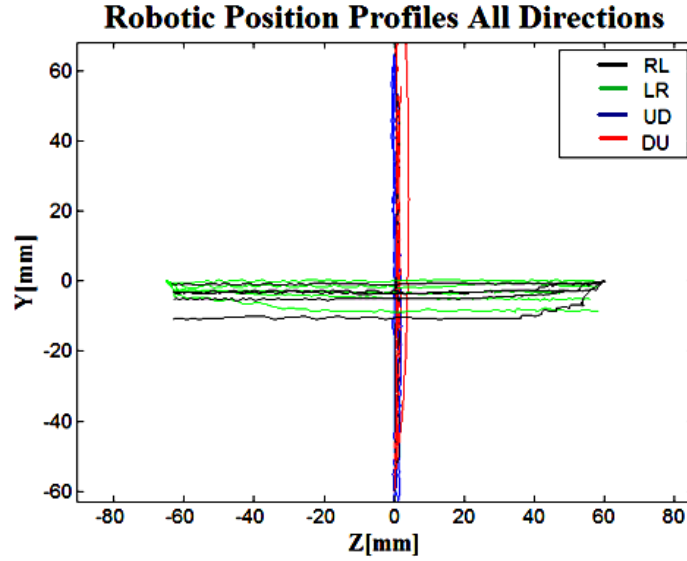


Figure 38: Robotic position profiles for all hybrid algorithms in all the directions.

Table 20: PS measure, robotic experiment. Measure range [-1:1]

Direction \ Algorithm	CS 1mm	CS 1.5mm	CS 2mm	P stepping	PD stepping
LR	0.9598	0.9453	0.9394	0.9529	0.9803
RL	0.9359	0.9282	0.9407	0.9714	0.9756
DU	0.9936	0.9969	0.9948	0.9976	0.9987
UD	0.9959	0.9943	0.9917	0.9920	0.9944

The OB measure shows that in the horizontal movements there is a clear tendency downwards (Figure 39), no such tendency was found in the vertical movements. The PD algorithm is very close to the straight-line projection in the frontal plane, but also has an OB measure which is negative, Table 21. This result is very similar to the result which was found in human subjects.

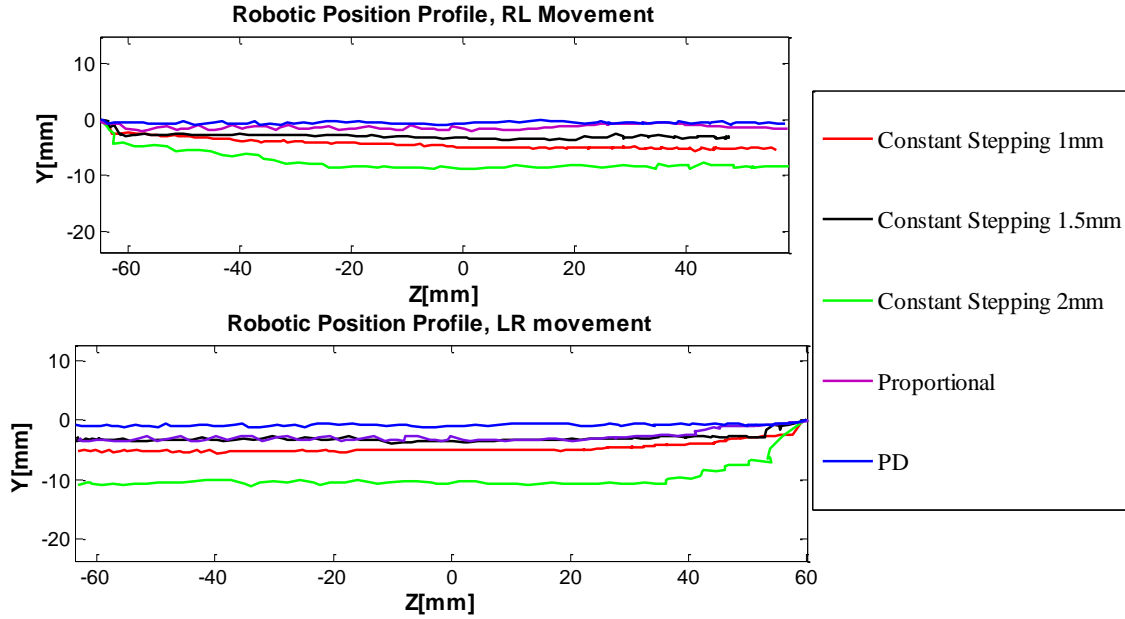


Figure 39: Robotic profiles - zoom in for LR and RL movements.

Table 21: OB measure, robotic experiment. No units, ranges from -1 to 1.

Algorithm \ Direction	CS 1mm	CS 1.5mm	CS 2mm	P stepping	PD stepping
LR	-0.9984	-1.0000	-0.9995	-0.9912	-0.9820
RL	-0.9942	-0.9823	-1.0000	-0.6897	-0.5589
DU	0.9996	0.9079	0.6266	0.9996	0.9976
UD	-1	0.7884	-0.1238	-0.1183	0.1575

4.5.2 ND, MD and FR Measures

The tradeoff between accuracy and precision is commonly encountered. Thus not surprisingly the results show (Table 22) that using a larger step is much faster in all movement direction (CS) but the number of detachments increases. The PD algorithm is almost as fast as the CS with 2mm step size but had no detachments while the CS with 2mm had between 3 to 4 detachments at each direction. Thus the PD was more precise and faster than most the other algorithms. When combining the two goals the PD algorithm is the most suitable for completing the task accurately and as fast as possible.

Table 22: ND and MD measures – robotic experiment.

Direction Algorithm	ND				MD			
	LR	RL	DU	UD	LR	RL	DU	UD
CS 1mm	1	0	1	0	24.5sec	23.9sec	20.7sec	21.8sec
CS 1.5mm	2	2	3	2	19.4sec	18.7sec	14.91sec	15.4sec
CS 2mm	3	3	4	4	12.6sec	12.6sec	11.55sec	11.5sec
P Stepping	0	0	0	0	16.5sec	15.9sec	14.59sec	14.4sec
PD Stepping	0	0	0	0	14.4sec	14.2sec	13.01sec	12.5sec

The FR which was applied in the robotic experiment show that the PD algorithm was once again the most stable in all movement direction, Table 23. In addition the forces in the vertical (DU, UD) movements were generally smaller than the forces in the horizontal (LR, RL) movements. It is clear from the results that for CS algorithms accuracy decrease with step size. The FR measure was smaller for CS with 1mm than for CS with 1.5mm and 2mm step size.

Table 23: FR measure – robotic experiment.

Direction Algorithm				
	LR	RL	DU	UD
CS 1mm	18.76N	18.68N	12.25N	12.92N
CS 1.5mm	22.16N	22.12N	13.28N	13.01N
CS 2mm	23.97N	24.11N	23.27N	23.11N
P Stepping	13.08N	12.9N	11.54N	11.27N
PD Stepping	11.27N	11.02N	9.99N	10.39N

4.6 Discussion

From the results it can be seen that the robotic motion profiles were similar to straight lines as previously seen in human motion profiles. It was shown that *hybrid* control is suitable for solving constrained motion on a hemisphere. The first interesting result was that a difference was found between vertical and horizontal movements. In the OB measure the horizontal movements showed a great tendency downwards similar to results of human profiles. All the horizontal movements (in the robotic experiment) started by moving downwards. The CS algorithm stabilized the error around the expected threshold. The PD and P control algorithms corrected the error fast enough to stabilize the position error close to zero and thus there was no need for adding the integral component.

The difference between the horizontal (LR, RL) and vertical (DU, UD) movements can be explained by the effect of the gravity on the motors. However there was also a difference in the motors that were activated during most of the movement in each direction. In the vertical movements the "shoulder" and the "elbow" (joint 2 and 3) perform the major part of the movement but in the horizontal movements the "base" creates most of the movement (joint 1, Figure 40). This may explain the strong difference in the range of forces when comparing the movement direction.

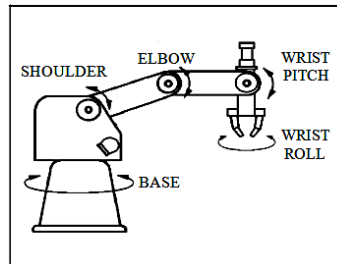


Figure 40: Scorebot Er 5+ structure.

The similarity that was found in the OB and PS measure may imply that it is feasible humans use a hybrid control scheme. However the available hardware limited performance and effected attainable similarity. The main problem was that the time of reaction was not fast enough ($\sim 0.16\text{sec}$) and thus the profiles were not clean enough for quantitative comparison (filtering could not solve this problem). Since that the velocity and force profiles were extremely noisy it should be considered that it is possible that using hybrid control with feedback as the main control loop is not the policy chosen by human subjects. Experiments done by McIntyre et al. (1995) suggested that humans are using feedforward rather than feedback control for constraint planar motion.

Regarding the best algorithm for the constrained motion task the results show that using PD stepping was better than the all other algorithms. The PD algorithm completed the task in the fastest

way and without any detachments. Even though the CS with the step size of 2mm is faster, this algorithm is less accurate than the PD algorithm in terms of the applied forces and in the ND measure. This finding was not surprising since it is known that the PD control is considered better than the rest of the algorithms which were tested.

Chapter Five: Conclusions and future research

In this research we tried to validate the use of a virtual environment and to test the effects of gravity on constrained motion on hemisphere. To reach those goals, a physical environment was developed and compared to the virtual environment and movement profiles in horizontal and vertical directions were examined. In addition we implemented constrained motion algorithm based on hypothesized human strategies using a robotic system to facilitate improved understanding of underlying planning and control mechanisms. Oblique motion profiles were studied in a parallel project.

For the validation of the virtual environment three main measures were compared: VM, MD and PS. The results showed that movements in the physical environment were significantly faster and significantly straighter. However, most of the differences had a physical explanation and it is possible that the virtual environment is simulating the physical environment, only not the one that was tested in our physical experiment. Creating a compatible physical environment is problematic, since it requires that the force sensor would be placed on the hemisphere. In addition it requires fixing the end point so that it wouldn't rotate. This could also be problematic and not very comfortable for the subject.

For testing the gravity effects two measures were defined: AD_{rel}/MD and OB. The results for the AD_{rel}/MD showed that asymmetries were found in horizontal movements, while in vertical movements the asymmetries were less significant. Those results were suitable to previous research on non-constrained motion (pointing movements) and in drawing movements (on planar surface).

The OB measure showed that horizontal movements had a large tendency downwards (towards the gravity) while vertical movements didn't have any of the horizontal directions. Although most of the subjects obeyed this observation, two out of ten subjects (each of them in a different direction) had a tendency upwards and thus this observation still needs to be validated in more experiments. In addition the results were attained in different environments and different speeds, which automatically translate to differences in the dynamics. Thus we suggest the notion that the CNS takes into account the gravity in the motion planning process.

The results of the robotic experiment indicate hybrid control is suitable for solving the problem of constrained motion on a hemisphere. In addition the robotic movements were close to straight line projections on a planar surface and the vertical movements had a large tendency downwards similar to the results in the human experiments. Thus it is encouraging to use robotics for researching human

constrained motion. The gravity effects which were found in the robotics motion profiles could be reasoned at partially by inertial manipulator characteristics.

This work examines fundamental issues in the field of human constrained motion. The findings can be generalized by examining additional and more complex surface types. Future work may use findings from the current work for applications such as development of objective motion measures and assistive robotics. The robotic implementation should be further developed in two directions:

- 1) Implement feed forward algorithms to help strengthen the understanding of differences between feed forward and feedback based motion planning on non planar surfaces.

- 2) Implementation with improved hardware. Improved hardware can reduce reaction time which was a major issue limiting the current implementation. In addition computer vision can be integrated in the motion planning and control algorithms strengthening the similarities between the human and robotic control systems.

References

- Flash .T, Hogan .N. "The Coordination of Arm Movements: An Experimentally Confirmed", The Journal of Neuroscience, 1985.
- D. Sha., Patton, J. L., Mussa-Ivaldi, F. A., "Minimum jerk reaching movements of human arm with mechanical constraints at the endpoint". *International Journal of Computers, Systems, and Signals*, 7, 41–50, (2006).
- D. G. Liebermann, T. Krasovsky, S. Berman, "Planning Maximally Smooth Hand Movements Constrained to Nonplanar Workspace", *Journal of motor behavior*, 2008, Vol 40, No.6, 516-531.
- E. Todorov, M. I. Jordan, "Smoothness maximization along a predefined path accurately predicts the speed profiles of complex arm movements". *Journal of Neurophysiology*, 80, 696–714, (1998).
- Uno .Y, Suzuki R. and Kawato M.. Minimum muscle-tension-change model which reproduces human arm movement. In *Proceedings of the 4th Symposium on Biological and Physiological Engineering*, (pp. 299–302). Fukuoka, Japan, 1989.
- Tanaka H, Tai M, and Qian N. Different predictions by the minimum variance and the minimum torque – change models on the skewness of movement velocity profiles. *Neural Comput* 12: 2021-2040, 2004.
- Yoshikawa.T, "Foundations of Robotics Analysis and Control", MIT Press Cambridge, Massachusetts, London. 1990.
- Spong .M.W Rapid Review of Kinematics. In M. W. Spong, *Robot Dynamics and Control* (pp. 2-35).
- Colgate .E and Hogan .B. "Robust control of dynamically interacting systems. *Int. J. Control* 48(1):65-88, 1988.
- M. Petrou and P. Bosdogianni, "Image Processing: The Fundamentals". John Wiley, 1999.
- Cheah .C.C and Wang .D , "Learning Impedance Control for Robotic Manipulators", *IEEE Transactions on Robotics and Automation*, VOL. 14, NO. 3, JUNE 1998.
- Bonitz.R.G and Hsia.T.C, "Robust Dual-arm Manipulation of Rigid Objects via Palm Grasping". *Proceedings of the IEEE International Conference on Robotics and Automation*. Vol. 4, pp.3047-3057, April 1996.
- Lawrence, D., and Stoughton, R. M., "Position-Based Impedance Control: Achieving Stability in Practice", *Proc. AIAA Conf. on Guidance, Navigation and Control*, 1987.
- Lawrence, D., "Impedance Control Stability Properties in Common Implementations", *Proc. IEEE conf. on Robotics and Automation*, 1988.
- T. Yoshikawa, "Dynamic hybrid position/force control of robot manipulators – description of hand constraints and calculation of joint driving force". *IEEE J. Robot. Autom.* 3, 386–392, 1987.
- Colgate .J.E and Hogan .N, "An analysis of contact instability in terms of passive physical equivalents," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1989, pp. 404-409.

- Hogan, N. 1985, "Impedance Control: An Approach to Manipulation". Part I-III. Journal of dynamic System, Measurement and Control vol.107.
- M.H. Raibert, J.J. Craig, "Hybrid position/force control of manipulators". ASME J. Dyn. Syst. Meas. Contr. 103, 1981.
- Serdar Kucuk and Zafer Bingul, "Robot Kinematics: Forward and Inverse Kinematics". Industrial Robotics Theory Modeling Control, ISBN 3-86611-285-8, pp. 964, ARS/plV, Germany, December 2006.
- Phosliri .C, Rabindran .D, Pryor .M and Kapoor .C, "Extended Generalized Impedance Control for Redundant Manipulators. 42nd IEEE Conference on Decision and Control, 2003.
- Ikeura.R, Inooka.H. "Variable Impedance Control of a Robot for Cooperation with a Human". IEEE international Conference on Robotics and Automation. 1995.
- Šurdilovic.D, "Synthesis of Impedance Control Laws at Higher Control Levels: Algorithms and Experiments". Proceedings of the IEEE, International Conference on Robotics and Automation, Leuven, Belgium, 1998.
- Šurdilovic.D, Kirchhof.J." A new Position Based Force/Impedance Control for Industrial Robots" Proceedings of the IEEE, International Conference on Robotics and Automation, Minneapolis, 1996.
- Valency.T, Zacksenhouse.M, "Instantaneous Model Impedance Control for Robots," Proceeding of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2000.
- Valency.T, Zacksenhouse.M, "Accuracy/Robustness Dilemma in Impedance Control," ASME Journal of Dynamics Systems and control, 310 Vol. 125, 2003.
- W. S. Newman, "Stability and performance limits of interaction controllers," ASME, J. of Dyn. Sys., Meas., Contr., vol.114,pp.563-570, 1992.
- Hogan.N, "stable execution of contact tasks using impedance control". in Proc, IEEE Int. Conf. Robotics and Automation, Raleigh, NC, pp. 1047-1054 ,1987
- Jagersand M. Nelson R. "Visual Space Task Specification, Planning and Control" In *Proc. IEEE Int. Symp. on Computer Ksion*, 1995.
- Bernstein N. "The coordination and regulation of movements". London: Pergamon, 1967.
- Kawato M., Furukawa K., & Suzuki R. A hierarchical neural-network model for control and learning of voluntary movement. *Biological Cybernetics*, 57, 169–185, 1987.
- Harris C. M., Wolpert D. M. Signal-dependent noise determines motor planning. *Nature*, 394, 780–784, 1998.
- Hamilton AF de C and Wolpert DM. Controlling the statistics of action: obstacle avoidance. *J Neurophysiol* 87: 2434-2440, 2002.
- Biess A, Lieberman DG, Flash T. A computational model for redundant human three – dimensional pointing movements: integration of independent spatial and temporal motor plans simplifies movement dynamics. *J Neurosci*, 2007.

Hashimoto .K. "*VISUAL SERVOING: Physical-Time control of Robot Manipulators Based on Visual Sensory Feedback*". World Scientific Series in Robotics and Automated System, World Scientific Press, Singapore, 1993.

Qiu L., Zhang H., Gao W., Ge Y. Vision/Force Based Robot Manipulator Servo Control for Uncertain Plane Surface Tracking. IEEE, 2008.

Danica Kragic and Henrik Christensen, "A Framework for Visual Servoing". *International Conference on Computer Vision Systems ,ICVS 2003*, Graz, Austria

Ezio Malis, F. Chaumette, and S. Boudet, "2-1/2-D Visual Servoing". *IEEE Trans. on Robotics and Automation*, Vol. 15, No. 2, pp.238-250,1999.

Walter, W. Grey, "An imitation of life". *Sci Am.* 182, 5 (May 1950), 42-45.

Samuel R. Buss, "Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods". Technical report, University of California, San Diego, USA. 2004.

L.-C. T. Wang and C. C. Chen, "A combined optimization method for solving the inverse kinematics problem of mechanical manipulators", *IEEE Transactions on Robotics and Automation*, 7 (1991), pp. 489-499.

C. W. Wampler, "Manipulator inverse kinematic solutions based on vector formulations and damped least squares methods", *IEEE Transactions on Systems, Man, and Cybernetics*, 16 (1986), pp. 93-101.

A. Balestrino, G. De Maria, and L. Sciavicco, "Robust control of robotic manipulators", in *Proceedings of the 9th IFAC World Congress*, Vol. 5, 1984, pp. 2435-2440.

D.E. Meyer, R.A. Abrams, S Kornblum, C.E. Wright, J.E.K. Smith, "Optimality in human motor performance: ideal control of rapid aimed movements". *Psychol Rev* 95:340–370, 1988.

N Dounskaia, D Wisleder, T Johnson, "Influence of biomechanical factors on substructure of pointing movements".*Exp Brain Res* 164:505–516, 2005.

D. Wisleder N. Dounskaia, "The role of different submovement types during pointing to a target", 2006.

Papaxanthis C, Pozzo T, Stapley P, "Effects of movement direction upon kinematic characteristics of vertical arm pointing movements in man". *Neurosci Lett* 253:103–106, 1998.

Papaxanthis C, Pozzo T, Schieppati M, "Trajectories of arm pointing movements on the sagittal plane vary with both direction and speed". *Exp Brain Res* 148(4):498–503, 2003.

Antonio C. Leite, Fernando Lizarralde and Liu Hsu, "Hybrid Adaptive Vision Force Control for Robot Manipulators Interacting with Unknown Surfaces". *The International Journal of Robotics Research* 2009; 28; 911.

Olsson, T., Bengtsson, J., Johansson, R. and Malm, H, "Force control and visual servoing using planar surface identification". *Proceedings IEEE International Conference on Robotics and Automation*, vol. 4, pp. 4211–4216, 2002.

Namvar, M. and Aghili, F, "Adaptive force–motion control of coordinated robots interacting with geometrically unknown environments". *Proceedings IEEE International Conference on Robotics and Automation*, vol. 3, pp. 3061–3068, 2004.

Karayiannidis, Y. and Doulgeri, Z, "An adaptive law for slope identification and force position regulation using motion variables". *Proceedings IEEE International Conference on Robotics and Automation*, pp. 3538–3543, 2006.

Soechting JF, Bueno CA, Herrmann U, Flanders M, "Moving effortlessly in three dimensions: Does Donders' law apply to arm movement?". *J Neurosci* 15:6271–6280, 1995.

Pellegrini JJ, Flanders M, "Force path curvature and conserved features of muscle activation". *Exp Brain Res* 110:80–90, 1996.

Soechting JF, Flanders M , "Movement planning: kinematics, dynamics, both or neither", pp 352–371. New York: Cambridge University Press, 1998.

Nishikawa KC, Murray ST, Flanders M, "Do arm postures vary with the speed of reaching?" *J Neurophysiol* 81:2582–2586, 1999.

Papaxanthis C, Pozzo T, McIntyre J, "Kinematic and dynamic processes for the control of pointing movements in humans revealed by short-term exposure to microgravity". *Neuroscience* 35(2):371–383, 2005.

R. Gentili, V. Cahouet, and C. Papaxanthis, "Motor Planning of Arm Movements Is Direction-Dependent in the Gravity Field," *Neuroscience* 145 (1), 20–32, 2007.

Papaxanthis, C., Dubost, V. & Pozzo, T. "Similar planning strategies for whole body and arm movements performed in the sagittal plane". *Neuroscience*, 117, 779–783, 2003.

Appendix A – Elaboration on the Pseudo inverse kinematics

The Jacobian matrix is defined as:

$$2.1. \quad J(q, \theta) = \begin{bmatrix} \frac{\partial q_1}{\partial \theta_1} & \frac{\partial q_1}{\partial \theta_2} & \cdots & \frac{\partial q_1}{\partial \theta_m} \\ \frac{\partial q_2}{\partial \theta_1} & \frac{\partial q_2}{\partial \theta_2} & \cdots & \frac{\partial q_2}{\partial \theta_m} \\ \vdots & \vdots & & \vdots \\ \frac{\partial q_n}{\partial \theta_1} & \frac{\partial q_n}{\partial \theta_2} & \cdots & \frac{\partial q_n}{\partial \theta_m} \end{bmatrix}$$

Where θ is a vector of the joint angles and which defines the position of the joint. Now the basic equation can be written as:

$$2.2. \quad \dot{q} = J(\theta)\dot{\theta} \rightarrow \Delta q \approx J(\theta)\Delta\theta$$

The question is how to choose $\Delta\theta$ and the main idea is to choose $\Delta\theta$ so that Δq will be close as possible to \vec{e} . One approach is to solve the following equation:

$$2.3. \quad \vec{e} = J(\theta)\Delta\theta$$

$$2.4. \quad \Delta\theta = J^{-1}(\theta)\vec{e}$$

Singularity of the Jacobian might lead to poor results and in many cases J is not invertible.

The *Jacobian Transpose* method (Balestrino et al., 1984) uses the transpose of the Jacobian matrix, instead of the inverse. Although J^T is not equal to J^{-1} , using it is justified experimentally by the use of virtual force (Buss, 2004).

The *Jacobian Transpose* equation is written as:

$$2.5. \quad \Delta\theta = \alpha \cdot J^T(\theta) \cdot \vec{e}$$

Where α should be chosen by minimizing the following expression:

$$2.6. \quad \alpha = \frac{\langle \vec{e}, J \cdot J^T \cdot \vec{e} \rangle}{\langle J \cdot J^T \cdot \vec{e}, J \cdot J^T \cdot \vec{e} \rangle}$$

The intuition for this minimization comes from the fact that for all J and \vec{e} $\langle \vec{e}, J \cdot J^T \cdot \vec{e} \rangle \geq 0$ since $\langle \vec{e}, J \cdot J^T \cdot \vec{e} \rangle = \langle J^T \cdot \vec{e}, J^T \cdot \vec{e} \rangle = \|J^T \cdot \vec{e}\|^2 \geq 0$.

The *Pseudo Inverse* method gives the best possible solution in the sense of least squares. The intuition for formulating this method comes from the fact that the equation of $J^T \cdot (\vec{e}) = z$, always has a solution (0). Using the previous equation and multiplying both sides we get:

$$2.7. \quad J^T (J \cdot \Delta\theta) = J^T \cdot (\bar{e})$$

This can be reformulated as:

$$2.8. \quad \Delta\theta = J^T (JJ^T)^{-1} \cdot (\bar{e})$$

The *Pseudo Inverse* is widely discussed in the literature, and it often leads to poor results because of its instability near singularities. The (*Selectively*) *Damped Least Squares* (SDLS) method shows better performances in obtaining the inverse Jacobian than the *Pseudo Inverse* method 0. Without going into all the details it uses the following equation to have a better solution:

$$2.9. \quad \Delta\theta = J^T (J^T J + \lambda^2 I)^{-1} \bar{e}$$

$\lambda \in \mathbb{R}$ is a non-zero damping constant, and is used to assure that $(J^T J + \lambda^2 I)$ would be invertible and without singularities.

Appendix B - PID Control

The *proportional–integral–derivative* controller (PID controller) is a basic control loop widely used in industrial control systems. The PID calculates the error between a measured variable and its desired value - $e(t) = X_{measured} - X_{desired}$. The goal of the controller is to minimize the error by correcting the process control inputs according to the measurements of the process. PID control can be used for position, velocity, acceleration and force control (which are all presented in the next sections), and the difference in the equations would be the error which is minimized.

PID is made up of three main components:

- 1) P (Proportional control) - The output varies is proportional to the error.

$$2.10. \quad P = k_p \cdot e(t).$$

where k_p is the proportional gain.

- 2) I (Integral control) - The output varies based on the sum of recent errors.

$$2.11. \quad I = k_i \cdot \int_0^t e(\tau) d\tau. \text{ } k_i \text{ is the integral gain.}$$

- 3) D (Derivative control) - The output varies based on the change in the error. Greater change is greater response, good for dampening spikes and jumps.

$$2.12. \quad D = k_d \cdot \frac{d}{dt} e(t). \text{ } k_d \text{ is the differential gain.}$$

In certain cases not all the components are needed, and then by setting the gain of undesired control outputs to zero it is possible to create simpler controllers such as PI, PD, P or I. It is common to define $u(t)$ as the controller output, and thus full equation of the PID algorithm (Figure 41) is:

$$2.13. \quad u(t) = k_p \cdot e(t) + k_D \cdot \frac{d}{dt} e(t) + k_i \cdot \int_0^t e(\tau) d\tau$$

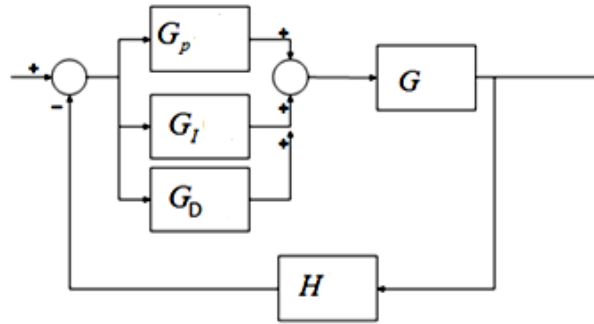


Figure 41: Block diagram of PID control loop.

G_p , G_I , G_D are the proportional, integral and differential blocks. G is the process and H represents the sensor.

Appendix C – Pre-processing – Reconstruction and Validation of the Tip

The reconstruction of the tip during the movement was based on the fact that the wooden plate and the tip were rigidly attached. First it was needed to reconstruct the translation and rotation of the wooden plate and then to reconstruct from it the tip.

The reconstruction was done by several basic steps. First the coordinate system of the Codamotion was defined as shown in Figure 42. Then the distance between the tip and the center of the three markers was found using one steady measurement from the Codamotion, outside of the hemisphere. That measurement was defined as the reference to all the other measures.

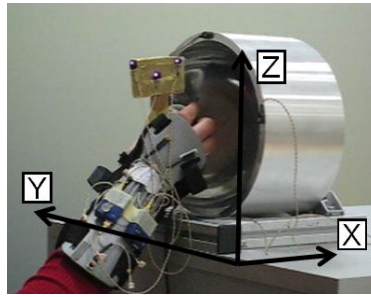


Figure 42: The Codamotion axis.

For the reconstruction P_0 , P_1 and P_2 were marked as the positions in space of the three markers (with x, y and z measures). Then for each sample recording the following stages were performed:

Two vectors were defined: $U_x = \frac{(P_2 - P_1)}{\|P_2 - P_1\|}$ and $U_y = \frac{(P_3 - P_1)}{\|P_3 - P_1\|}$. Those vectors were not orthogonal as shown in Figure 43. Using the cross product of $\frac{(U_x) \times (U_y)}{\|(U_x) \times (U_y)\|}$, U_z was defined and was orthogonal for both. Using the cross product of U_x and U_y . The final step was to define the correct third axis by: $U_y = \frac{(U_z) \times (U_x)}{\|(U_z) \times (U_x)\|}$ and to receive a right handed system.

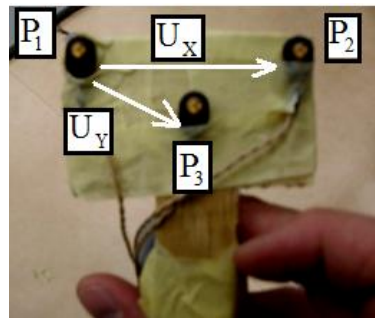


Figure 43: Defining the new transformation axes

The center of the three markers was found and marked as (O_x, O_y, O_z).

The total transformation matrix was defined by rotation and translation:

$$T = \begin{bmatrix} U_{x1} & U_{x2} & U_{x3} & O_x \\ U_{y1} & U_{y2} & U_{y3} & O_y \\ U_{z1} & U_{z2} & U_{z3} & O_z \\ 0 & 0 & 0 & 1 \end{bmatrix} ; \quad Tip_0 = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \\ 1 \end{bmatrix}$$

U was defined in above and sets the rotation of the wooden plate. O sets the translation and Tip_0 is the original distance between the tip and the center of the three markers.

Once U was found the new position of the tip was calculated by the following transformation:

$$T_{New[4 \times 1]} = U_{[4 \times 4]} \cdot Tip_{0[4 \times 1]}$$

Validation

In order to validate that the reconstruction is accurate two tests were done:

A fourth marker was placed on the wood, and thus its position during a movement was recorded, Figure 44. Then reconstruction of the fourth marker was compared to the results from the fourth marker.

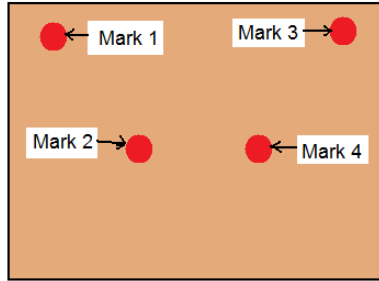


Figure 44: Setup for testing the reconstruction

Figure 45 shows the reconstruction of the fourth marker which. The Green line is the original and actual position in the space, and the Blue is the reconstruction. The accuracy which was tested in smaller resolutions was 0.5mm in worst case.

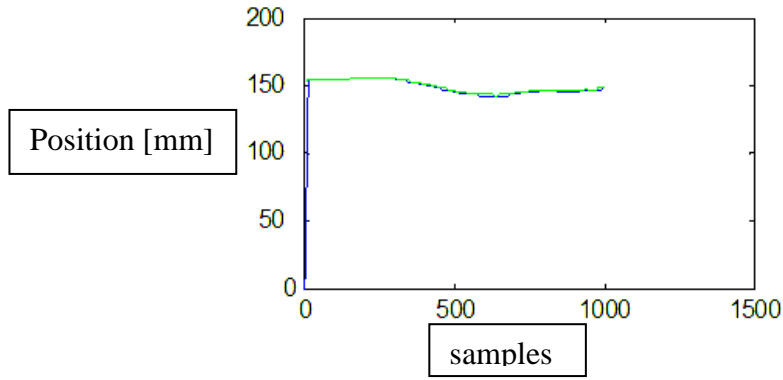


Figure 45: Reconstruction of a fourth marker, green original movement, blue reconstruction.

In order to make sure that our good results are not affected by the fact that the fourth marker was placed on the same wood but the tool end point was in an orthogonal axis, we plotted all the movements of several subjects in expectation to receive a hemisphere shape.

Figure 46 shows two plots. The Red is the reconstruction of the tip, and the Cyan is the data from one of the markers. The marker moved in all the directions due to rotations of the hand, and thus created a full view of the hemisphere. The reconstruction of the tip actually shows the straight lines which were created during the experiment. In addition we can notice that the diameter is approximately 250mm=25cm as expected.

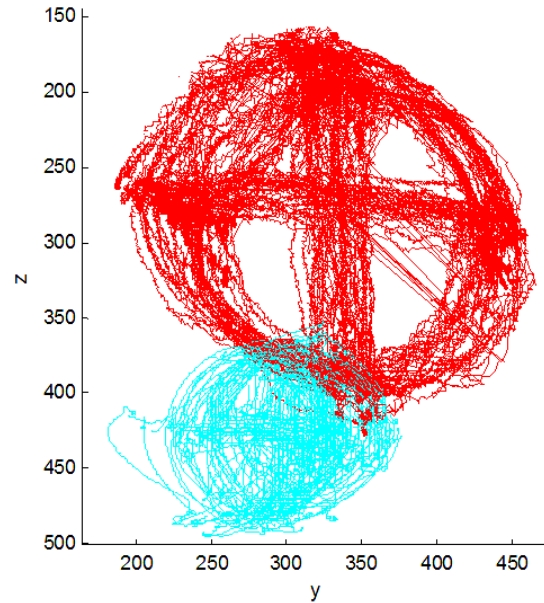


Figure 46: Plotting all the movements for one subject.

Appendix D- Full Post Hoc analysis - OB measure for the horizontal movements

Table 24: Post hoc analysis – OB measure – LR movements

Source	Subject 1							
Subject	2	3	4	5	6	7	9	10
P-value	.111	.728	.890	.002	.106	.578	.255	.111
Source	Subject 2							
Subject	1	3	4	5	6	7	9	10
P-value	.111	.172	.169	.000	.984	.317	.515	.999
Source	Subject 3							
Subject	1	2	4	5	6	7	9	10
P-value	.728	.172	.861	.000	.165	.792	.396	.172
Source	Subject 4							
Subject	1	2	3	5	6	7	9	10
P-value	.890	.169	.861	.003	.163	.695	.364	.169
Source	Subject 5							
Subject	1	2	3	4	6	7	9	10
P-value	.002	.000	.000	.003	.000	.001	.000	.000
Source	Subject 6							
Subject	1	2	3	4	5	7	9	10
P-value	.106	.984	.165	.163	.000	.307	.500	.983
Source	Subject 7							
Subject	1	2	3	4	5	6	9	10
P-value	.578	.317	.792	.695	.001	.307	.635	.317
Source	Subject 9							
Subject	1	2	3	4	5	6	7	10
P-value	.255	.515	.396	.364	.000	.500	.635	.515
Source	Subject 10							
Subject	1	2	3	4	5	6	7	9
P-value	.111	.999	.172	.169	.000	.983	.317	.515

Table 25: Post hoc analysis – OB measure – LR movements, virtual environment

Source	Subject 1								
Subject	2	3	4	5	6	7	8	9	10
P-value	.962	.245	.264	.673	.174	.000	.442	.857	.858
Source	Subject 2								
Subject	1	3	4	5	6	7	8	9	10
P-value	.962	.262	.282	.639	.188	.000	.469	.824	.830
Source	Subject 3								
Subject	1	2	4	5	6	7	8	9	10
P-value	.245	.262	.966	.131	.930	.000	.656	.231	.274
Source	Subject 4								
Subject	1	2	3	5	6	7	8	9	10
P-value	.264	.282	.966	.143	.894	.000	.688	.247	.290
Source	Subject 5								
Subject	1	2	3	4	6	7	8	9	10
P-value	.673	.639	.131	.143	.083	.000	.247	.853	.888
Source	Subject 6								
Subject	1	2	3	4	5	7	8	9	10
P-value	.174	.188	.930	.894	.083	.000	.564	.173	.220
Source	Subject 7								
Subject	1	2	3	4	5	6	8	9	10
P-value	.000	.000	.000	.000	.000	.000	.000	.000	.000
Source	Subject 8								
Subject	1	2	3	4	5	6	7	9	10
P-value	.442	.469	.656	.688	.247	.564	.000	.398	.443
Source	Subject 9								
Subject	1	2	3	4	5	6	7	8	10
P-value	.857	.824	.231	.247	.853	.173	.000	.398	.984
Source	Subject 10								
Subject	1	2	3	4	5	6	7	8	9
P-value	.858	.830	.274	.290	.888	.220	.000	.443	.984

Appendix E - Scorebot Er 5+ specifications

The robotic manipulator (Scorebot Er 5+) includes six degrees of freedom, and thus 6 engines. Each engine can create a different type of movement by itself (engines 1-3, 6) or using another engine (4,5), Table 26 and Figure 47, and Figure 48.

Table 26: Scorebot Er 5+ engines.

Engine #	Movement Description	Axis
1	Rotation of the manipulators body	Base
2	Lifting and lowering of the forearm	Shoulder
3	Lifting and lowering of the arm	Elbow
4+5	Lifting and lowering of the gripper	Gripper deviation
4-5	Rotation of the gripper	Gripper rotation
6	Opens and closes the gripper	

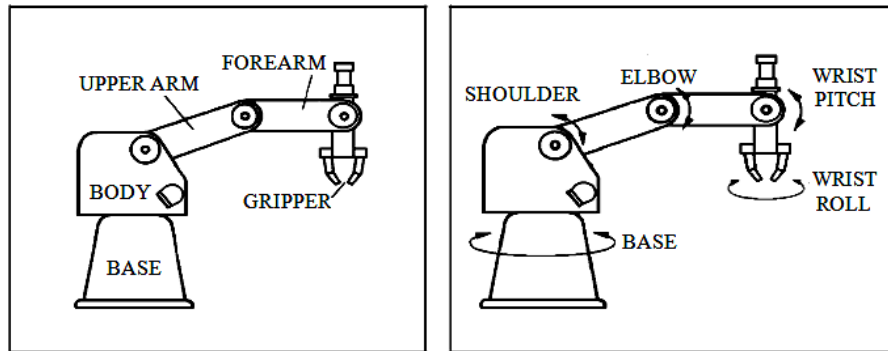


Figure 47: Scorebot Er 5+ structure, from the Intelitek Inc user manual

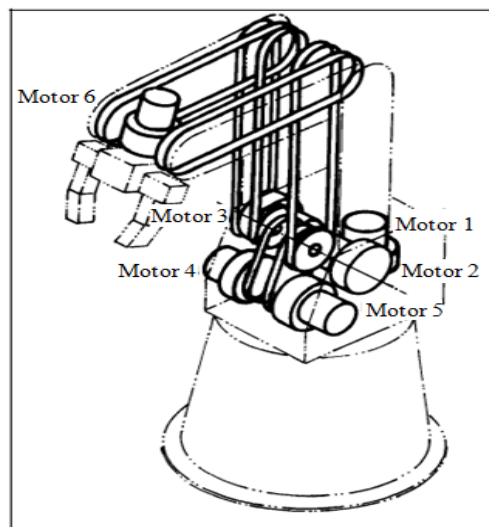


Figure 48: Scorebot Er 5+ engines, from the Intelitek Inc user manual

The gripper operation mode is differential. The gripper has three cog-wheels. Two of them are attached to engines four and five, by a set of stripes. When engine four and five are twisted in the same direction the gripper would go up and down and when the engines are twisted in opposite directions the gripper would rotate around it (the rotation axis goes through the center of the gripper using the third wheel, Figure 49).

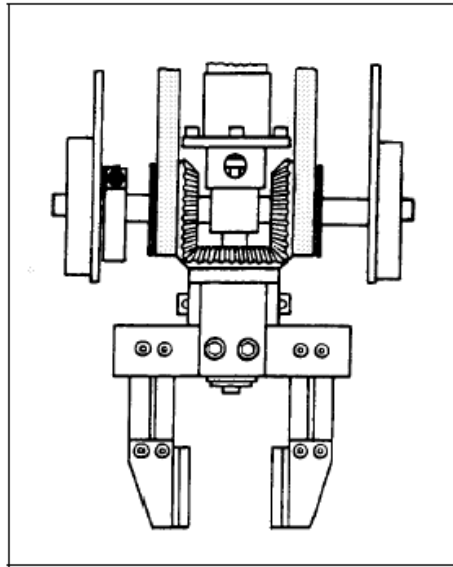


Figure 49: Overview on the gripper, from the Intelitek Inc user manual

Appendix F - Forward Kinematics and Jacobian for Scorbob ER 5+

The Scorbob Er 5 engines can work separately, thus an action in engine number 2 shouldn't affect the orientation of engines 3-6 but it would affect their position in space.

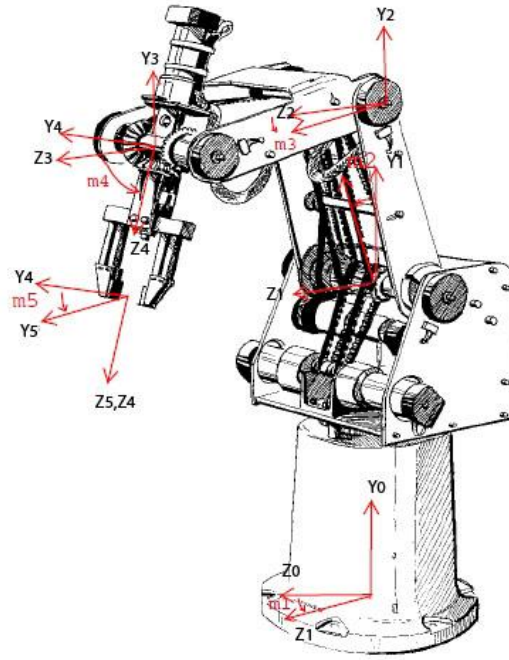


Figure 50: Scorbob ER 5, with six degrees of freedom, from the Intelitek Inc user manual.

The forward kinematics is according to the markings in Figure 50:

- 1) Rotation around angle m_1 , and translation around L_1 (\hat{y}):

$${}^0_1A = \begin{bmatrix} C1 & 0 & S1 & 0 \\ 0 & 1 & 0 & 0 \\ -S1 & 0 & C1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & L1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- 2) System 2 – to system 1

We have rotation in m_2 (\hat{x}), then translation to L_2 (\hat{y}) and then another rotation backwards of $-m_2$ (\hat{x}):

$${}^1_2A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C2 & -S2 & 0 \\ 0 & S2 & C2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & L2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C2 & S2 & 0 \\ 0 & -S2 & C2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & L2C2 \\ 0 & 0 & 1 & L2S2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3) System 3 – to system 2: Similar to the previous transformation:

$${}^2_3A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C3 & -S3 & 0 \\ 0 & S3 & C3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C3 & S3 & 0 \\ 0 & -S3 & C3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

4) System 4 – to system 3: rotation around \hat{x}_4 :

$${}^3_4A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C4 & -S4 & 0 \\ 0 & S4 & C4 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

5) System 5 – to system 4: Rotation $(\hat{Z}, -m5)$ plus translation (\hat{z}, L_4) :

$${}^4_5A = \begin{bmatrix} C4 & S4 & 0 & 0 \\ -S4 & C4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

6) All together we get: ${}^0_5A = {}^0_1A \cdot {}^1_2A \cdot {}^2_3A \cdot {}^3_4A \cdot {}^4_5A$

$${}^0_5A = \begin{bmatrix} C1C5 - S1S4S5 & C5S1S4 + C1S5 & C4S1 & S1(L3C3 + L4C4 + L2S2) \\ -C4S5 & C4C5 & -S4 & L1 + L2C2 - L3S3 - L4S4 \\ -C5S1 - C1S4S5 & C1C5S4 - S1S5 & C1C & C1(L3C3 + L4C4 + L2S2) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Jacobian –

For the Jacobian calculations there is only need for the end point and thus the derivatives are calculated for X, Y, and Z.

$$J = \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix} = \begin{bmatrix} C1(L3C3 + L4C4 + L2S2) & S1L2C2 & -S1L3S3 & -S1L4S4 \\ 0 & -L2S2 & -L3C3 & -L4C4 \\ -S1(L3C3 + L4C4 + L2S2) & -C1L3S3 & -C1L3S3 & -L4S4 \end{bmatrix}$$

The motor commands are given by

$$q_d = L^{-1}(x_{im}(t + \Delta t)) \quad ; \quad \dot{q}_d = J^{-1}\dot{x}_{im}(t + \Delta t)$$

Appendix G - Inverse Kinematics for Scorbot ER 5+

The inverse kinematics which will be presented here has the following limitations:

Angle m_1 (engine 1) must remain in the range of $-\frac{\pi}{2} < m_1 < \frac{\pi}{2}$. The arm is always considered in an upper elbow, Figure 51. There are more possible solutions for the system but they are not relevant for the experiment.

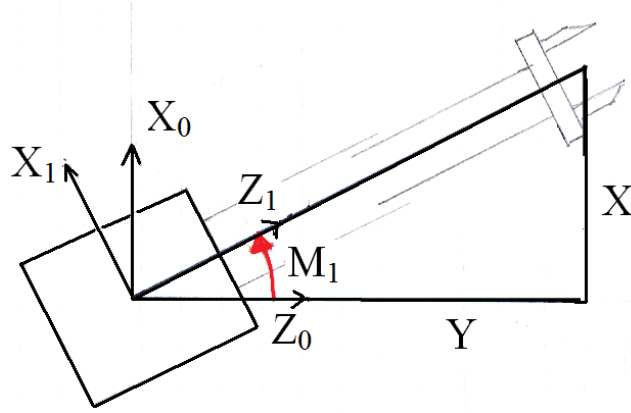


Figure 51: Axes for the calculation of engine 1 (m_1)

$$m_1 = A \tan\left(\frac{X}{Z}\right)$$

For angles m_2 , m_3 :

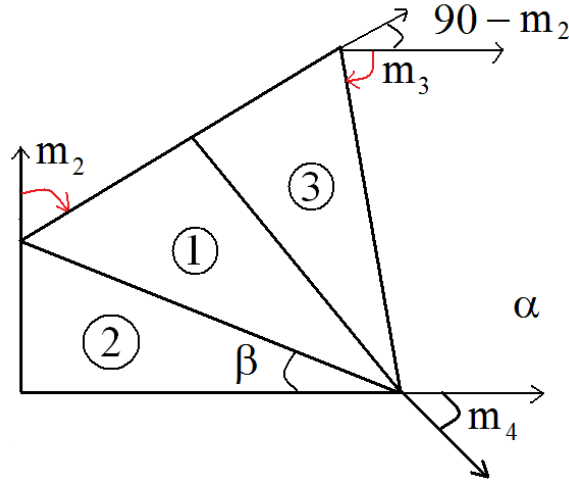


Figure 52: Axes for the calculation of engines 2 and 3 (m_2 , m_3)

From triangles 1 and 3 we can see that:

$$\begin{aligned} & \left(\sqrt{X^2 + Y^2} - L4 \cdot \cos(m4) \right)^2 + (L1 - L4 \cdot \sin(m4) - Y)^2 = \\ & = L2^2 + L3^2 - 2 \cdot L2 \cdot L3 \cdot \cos(180 - (m3 + 90 - m2)) \end{aligned}$$

For simplicity of the markings we define: $\gamma = m3 + 90 - m2$ and thus we receive:

$$\gamma = m3 + 90 - m2 = \cos^{-1} \left(\frac{\left(\sqrt{X^2 + Y^2} - L4 \cdot \cos(m4) \right)^2 + (L1 - L4 \cdot \sin(m4) - Y)^2 - (L2^2 + L3^2)}{2 \cdot L2 \cdot L3} \right)$$

Assuming γ is given, than α could be found from triangle 1 by:

$$\alpha = \tan^{-1} \left(\frac{L3 \cdot \sin(\gamma)}{L2 + L3 \cdot \cos(\gamma)} \right)$$

From triangle 2 β could be found by:

$$\beta = \tan^{-1} \left(\frac{L1 - L4 \cdot \sin(m4) - Y}{\sqrt{X^2 + Y^2} - L4 \cdot \cos(m4)} \right)$$

Angles m_2, m_3 are defined by:

$$m2 = \frac{\pi}{2} - \alpha + \beta$$

$$m3 = \gamma + m2 - \frac{\pi}{2}$$

To summarize the calculations we found the first three angles by:

$$\gamma = \cos^{-1} \left(\frac{\left(\sqrt{X^2 + Y^2} - L4 \cdot \cos(m4) \right)^2 + (L1 - L4 \cdot \sin(m4) - Y)^2 - (L2^2 + L3^2)}{2 \cdot L2 \cdot L3} \right)$$

$$m1 = A \tan \left(\frac{X}{Z} \right)$$

$$m2 = \frac{\pi}{2} - \tan^{-1} \left(\frac{L3 \cdot \sin(\gamma)}{L2 + L3 \cdot \cos(\gamma)} \right) + \tan^{-1} \left(\frac{L1 - L4 \cdot \sin(m4) - Y}{\sqrt{X^2 + Y^2} - L4 \cdot \cos(m4)} \right)$$

$$m3 = \gamma + m2 - \frac{\pi}{2}$$

A MATLAB code was written for the inverse kinematics and was used in the robotic hybrid control application.

Appendix H - ATI mini 45 – Force Sensor specifications

The setup included an ATI Mini 45 force sensor, Figure 53. The sensor was selected due to its high endurance to forces, and thus to avoid harmful situations in which the impact of the sensor \ manipulator with the surface might harm the sensor. The sensor can take up to $\pm 10000\text{N}$ in Z axis, and $\pm 5100\text{N}$ in X and Y axis,

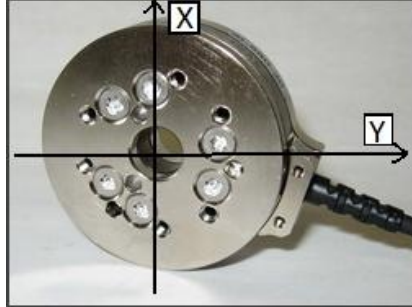


Figure 53: The axis of Ati Mini 45 force sensor

The calibration matrix (transferring volts into forces and torques) for the force sensor is:

CaliMat=[0.79508 -0.06630 -4.97248 31.8407 2.75546 -37.5483
4.48963 -37.8594 -2.32447 18.4244 -2.39372 21.6777
-18.5733 0.66213 -19.2161 0.36265 -18.6277 0.10539
0.37628 -0.07474 34.7814 -0.65913 -34.0970 0.47792
-38.1604 1.39559 19.9940 -0.45689 20.4380 -0.13513
-1.98154 17.8468 -2.41389 17.3101 -1.75935 20.57352];

This matrix also needs to be normalized by dividing it with the following norm vector (shown here only in 4 digits accuracy):

Norm=[0.3594 0.3594 0.1441 16.0027 16.0027 13.2208];

The normelization scheme is as follows:

- 1) CaliMat=CaliMat/Norm
- 2) Sampling the force sensor for a real data
- 3) Multiplying the normalized calibration matrix (1) with the data:

Forces = CaliMat · data

Appendix I - Row Data Human Experiments

In each experiment the data was recorded using different position and force sensors to a TXT file. Table 27 shows an example of raw data acquired in the virtual environment experiment. The data included the sample number, time, phase (stage in the process as explained in experiment), positions and forces in X, Y, and Z coordinates, the position of the target (TargX...), number of the trial and the target serial ID (each target received a number from 1 to 9).

The number of rows in each TXT table changed according to the experiment and recording time, but varied between 15000 to 23000.

Table 27: Acquired data from the virtual environment.

	Time	Phase	PosX	PosY	PosZ	FrcX	FrcY	FrcZ	TargX	TargY	TargZ	Trial	TrgID
0	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
1	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
2	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
3	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
4	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
5	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
6	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
7	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
8	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
9	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
10	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
11	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
12	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
13	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
14	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
15	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
16	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
17	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
18	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
19	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
20	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5
21	185.31	0	-0.47	0.74	-0.04	1.81	-2.80	0.14	-0.48	0.82	0	27	5

Appendix J – Analyzed Data

The TXT files were analyzed and then transferred to excel files. From the excel files we then moved them to SPSS. The following table shows part of the SPSS file which includes the results of the measures for each subject. The presented data includes only part of the data due to space saving. The entire data appears in the added Disk and supplements.

Table 28: Analyzed data in the SPSS.

	subject	MOD	Line Len	Curve Len	PS	OB	ID_Num	ID_Name	error	age	Virtual Or Physical
1	4	0.10	0.00	0.00	65,500	#NULL!	0.00	RL	5	30	1.00
2	1	0.08	1.50	1.52	0.99	0.89	5.00	UD	0	39	1.00
3	1	0.06	1.55	1.56	0.99	-1.00	5.00	UD	0	39	1.00
4	1	0.06	1.59	1.61	0.98	0.29	5.00	UD	0	39	1.00
5	1	0.06	1.52	1.54	0.99	-0.46	5.00	UD	0	39	1.00
6	1	0.09	1.56	1.58	0.99	-0.62	5.00	UD	0	39	1.00
7	2	0.12	1.55	1.70	0.91	0.64	5.00	UD	0	39	1.00
8	2	0.16	1.55	1.58	0.98	0.36	5.00	UD	0	55	1.00
9	2	0.10	1.52	1.67	0.91	-0.81	5.00	UD	0	55	1.00
10	2	0.08	1.55	1.57	0.99	-0.91	5.00	UD	0	55	1.00
11	2	0.11	1.54	1.59	0.97	-0.54	5.00	UD	2	55	1.00
12	3	0.05	1.52	1.52	1.00	-0.80	5.00	UD	0	31	1.00
13	3	0.15	1.62	1.70	0.95	0.93	5.00	UD	0	31	1.00
14	3	0.11	1.51	1.54	0.98	-1.00	5.00	UD	0	31	1.00
15	3	0.09	1.54	1.57	0.99	-0.34	5.00	UD	0	31	1.00
16	3	0.15	1.57	1.61	0.97	-1.00	5.00	UD	0	31	1.00
17	4	0.15	1.19	1.26	0.95	-0.91	5.00	UD	1	30	1.00
18	4	0.57	1.51	1.99	0.76	-1.00	5.00	UD	0	30	1.00
19	4	0.17	1.51	1.56	0.97	-0.77	5.00	UD	0	30	1.00
20	4	0.12	1.50	1.58	0.95	-0.90	5.00	UD	3	30	1.00

Appendix K – Raw Data Robotic Experiment

The TXT files were analyzed and then transferred to excel files. From the excel files we then moved them to SPSS. The following table shows part of the SPSS file which includes the results of the measures for each subject. The presented data includes only part of the data due to space saving. The entire data appears in the added Disk and supplements.

Table 29: Analyzed data in the SPSS.

	X Pos [mm]	Y Pos [mm]	Z Pos [mm]	Force X [N]	Force Y [N]	Force Z [N]	Time [sec] (delta T)
1	453.22	201.84	86.53	2.15	-20.80	5.27	0.171
2	453.63	202.06	86.61	2.30	-20.32	4.15	0.160
3	454.78	203.32	86.77	2.44	-19.97	-1.02	0.160
4	455.00	204.94	86.87	2.55	-21.05	3.36	0.160
5	454.29	206.08	86.62	2.19	-21.34	8.17	0.160
6	455.21	208.19	86.80	2.04	-21.24	5.67	0.160
7	456.19	209.38	86.98	2.36	-21.32	5.23	0.160
8	457.15	211.42	87.11	1.80	-20.96	4.56	0.160
9	457.99	213.48	87.15	2.81	-21.38	4.61	0.160
10	458.99	215.43	87.28	2.25	-21.00	3.32	0.160
11	459.89	217.47	87.03	2.37	-21.27	3.29	0.160
12	460.83	219.48	87.15	2.31	-20.73	3.17	0.160
13	461.68	221.45	87.25	2.26	-20.81	1.53	0.160
14	462.36	223.28	87.26	2.06	-20.48	-2.16	0.160
15	462.70	224.89	87.39	3.16	-20.29	-2.46	0.160
16	461.62	225.65	87.18	2.34	-20.76	2.61	0.160
17	461.29	227.16	86.94	1.91	-21.37	8.21	0.160
18	462.06	229.30	87.03	2.08	-21.22	5.59	0.160
19	462.85	231.05	87.18	2.16	-21.24	2.34	0.160
20	463.84	233.17	87.36	2.12	-20.63	-1.37	0.160

Appendix L – Percentages analysis of the Outlier removal

Virtual environment:

Table 30: Outlier for all Movements, virtual environment

Outlier Type	Total	Percentage from
No Error	907	79.28%
Early start	37	3.23%
Zero Force	193	16.87%
Other	7	0.61%
Total	1144	100.00%

In Table 30 we can see that 79.28% of all the measured movements had no error during the experiment. 16.87% of the movements had zero force and thus were neglected.

Table 31: Distribution of outliers per subject.

Outlier Type	Subject Number										
	1	2	3	4	5	6	7	9	9	10	11
No Outlier	94	96	97	78	104	81	20	65	76	95	101
Early start	6	2	3	8	1	2	1	40	28	9	4
Zero Force	0	2	4	18	1	21	82	0	0	0	0
Other	4	0	0	0	0	0	1	0	0	0	0
Total	104	100	104	104	106	104	104	105	104	104	105

Table 32: Percentages distribution of outliers per subject.

Outlier Type	Subject Number										
	1	2	3	4	5	6	7	9	9	10	11
No Outlier	90.3%	96.0%	93.2%	75%	98.12%	77.9%	19.2%	61.9%	73.1%	91.3%	96.2%
Early start	3.8%	2.0%	2.8%	7.7%	0.94%	1.9%	1%	38.1%	26.9%	8.7%	3.8%
Zero Force	0%	2.0%	3.8%	17.3%	0.94%	20.2%	78.8%	0%	0%	0%	0%
Other	3.8%	0%	0%	0%	0%	0%	1%	0%	0%	0%	0%
Total	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

Most of the Outliers came from subject number 7. Without this problem the accuracy percentage rises from 79.28% to 85.2%.

Errors - physical experiment:**Table 33:** Outlier for all Movements, physical environment

Error Type	Total	Percentage from
No Error	263	91.32%
Early start	1	0.35%
Zero Force	0	0%
Other	24	8.33%
Total	288	100%

Table 34: Distribution of outliers per subject.

Error Type	Total	Percentage from
No Error	88	91.67%
Early start	0	0%
Zero Force	0	0.00%
Other	8	8.33%
Total	96	100%

Table 35: Distribution of outliers per subject.

Error Type	Subject Number			
	1	2	3	4
No Error	72	70	63	58
Early start	0	0	0	1
Zero Force	0	0	0	0
Other	0	2	9	13
Total	72	72	72	72

Table 36: Percentages distribution of outliers per subject.

Error Type	Subject Number			
	1	2	3	4
No Error	100%	97.22%	87.5%	80.56%
Early start	0%	0%	0%	1.39%
Zero Force	0%	0%	0%	0%
Other	0%	2.88%	12.5%	18.06%
Total	100%	100%	100%	100%

Appendix M – ReadMe file

The supplemented disk has all the work that was done sorted into folders. The disk includes the following ReadMe file, which explains which materials could be found on the disk and how to operate several of the functions which were placed on the disk.

In the following contents each number refers to a folder, sub folder or a movie. In the following page, there is a brief explanation on each folder and its contents.

Contents of the Disk:

1. Physical Experiment – this subfolder includes all the work that was done in physical environment.

1.1. TXT - Physical Data – includes all the data that was recorded in TXT files.

1.2. MATLAB Segmentation – includes all the MATLAB files which were used for the segmentation process. The Gui for running the semi-automatic segmentation is activated by: "MainMenu.MAT" (which appears in this subfolder). In addition at the end of this readme file, there is manual user for the segmentation.

1.3. MATLAB – Results – This subfolder contains all the data manipulations which were done. In order to activate the main process, open "Results.MAT", and press F5. The rest is done mostly automatically.

2. Virtual Experiment – The folder is very similar to the physical folder, with one main difference: all the TXT files were compressed in order to save disk space. The rest is the same, in order to activate the files one needs to open the compressed TXT files, otherwise it would not work. Another difference is that in "**MATLAB – Results**" subfolder, the main function is "ReadingXLS".

3. Robotic Experiment – This folder includes the programming of the robot and the analyzing of the robotic profiles.

3.1. Basic Functions – This subfolder includes the basic functions for operating and communicating with the Scorebot Er 5.

3.2. MATLAB Programming – This subfolder includes the Hybrid control algorithms which were developed and some basic functions which were used such as homing and inverse kinematics.

3.3. Data analyzing – This subfolder contains four subfolders, one for each movement that was tested. In each subfolder there is a main function by the name: "Robotic profiles XX". This function runs all the calculations which were done for analyzing the data. The data is stored as MATLAB matrix type, and thus it is loaded automatically, through the function.

4. Movies – In order to emphasize the work that was done, three movies were created:

4.1. Building the Physical Environment – which documents the work that was done in Paris, including the laboratory and the construction of the system.

4.2. Robotic Constrained motion – documents the work (mostly the results) that was done in the mechanical engineering at BGU University.

4.3. Virtual Environment – documents the experiment in the virtual environment.

5. SPSS – All the statistical work which was done in SPSS is sorted in the "SPSS" folder. It includes two subfolders: "Data" and "Measures".

5.1. Subfolder "Data" includes two files, one which includes the data for the AD/MD measure, and one for all the rest of the measures.

5.2. Subfolder "Measures" includes the performance measures of the human experiments. It includes all the plots for all the measures, sorted by meaning full names, e.g. the PS measure has its own plot file and etc...

6. References – All the references which were used for this book are included in the "References" folder, divided by subcategories with meaning full names, e.g. hybrid, impedance and etc...

- User Manual for the MATLAB segmentation with a thorough explanation of how to use the semi automatic segmentation is given in the next pages.

Appendix N – User Manual for the MATLAB segmentation

This section explains how to use the segmentation semi automatic process which was developed for this thesis. Since the physical and virtual semi automatic procedures were very similar the procedure is explained only for the physical environment.

In order to use it enter the **"Physical Experiment"** folder (in the given disk). Than go to "MATLAB Segmentation" subfolder, open "MainMenu.mat" and press F5. This should open the MATLAB and run the MATLAB Gui (Figure 54).

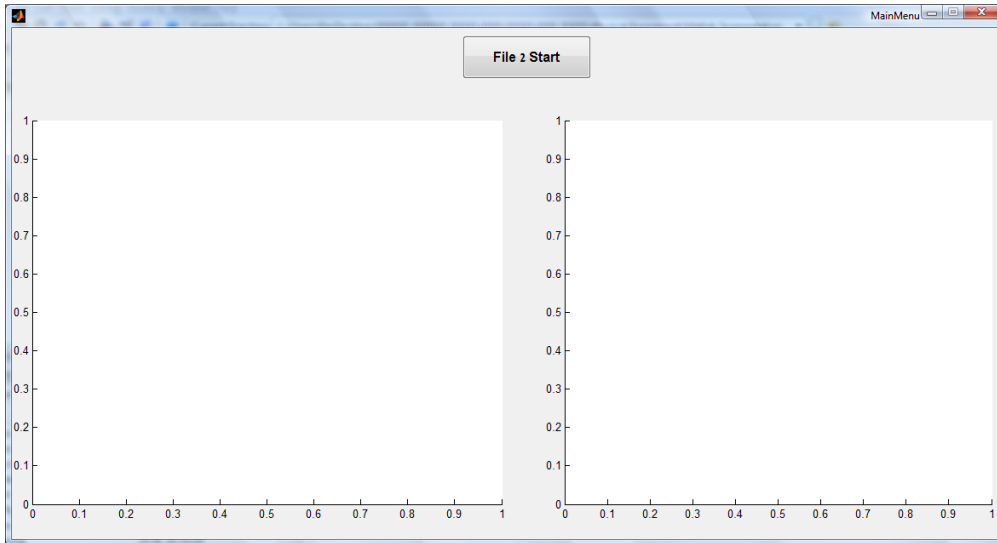


Figure 54: Opening screen of the Gui.

Press the "File 2 Start" button, and it would browse your computer (Figure 55).



Figure 55: The browse screen.

Enter the "TXT - Physical Data" select one of the subjects and then select the first TXT file. This would start the segmentation process and the previous Gui image would now have a force and velocity profile on it (Figure 56).

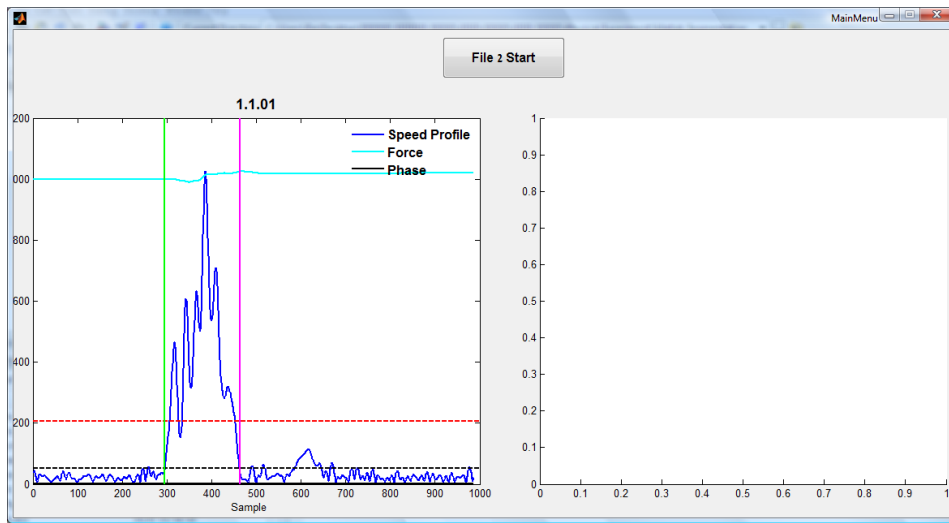


Figure 56: Force and velocity profiles on the Gui.

The green line signals the position in which the measurements begin and the pink line signs the end. You can move the lines using your mouse, and thus you can change the process. You could notice that in most cases the pink and the green lines would be located exactly where you would them to be for the segmentation, and that is why it is semi automatic. A question would appear on your screen (Figure 57).

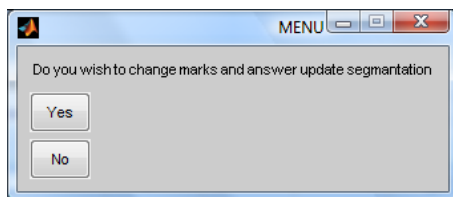


Figure 57: Do you wish to change marks and update segmentation question.

By saying yes, you change the start and stop positions of the measured data according to your selection of the segmentation. By saying no the start and stop of the data would remain as the default selection. On the right side of the Gui you would receive the movement which was made on the hemisphere Figure 58.

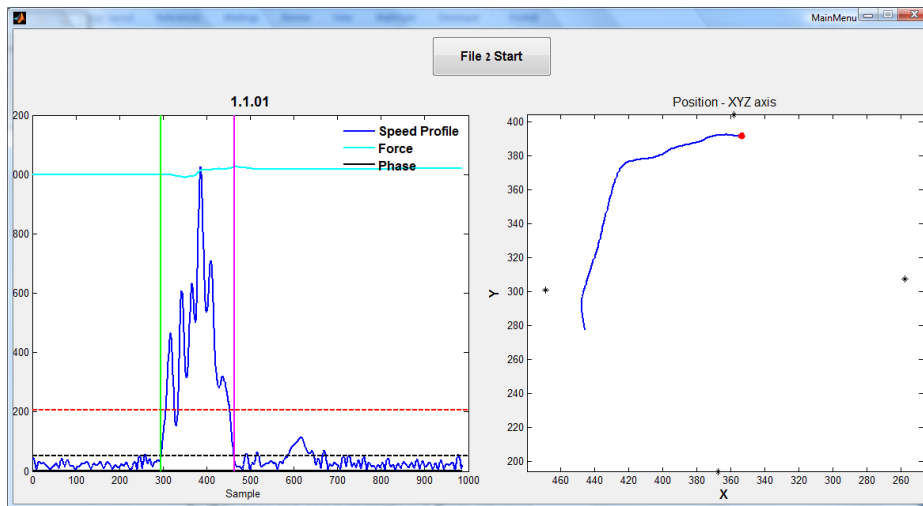


Figure 58: The addition of the position profile.

In this case we see a movement from the upper side of the hemisphere to the left side of the hemisphere. The red point is the sign for the beginning of the movement. If you would like to correct the segmentation (due to wrong position profile) then click yes on the next question (Figure 59).

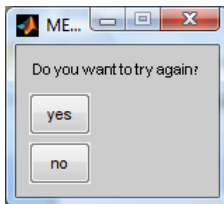


Figure 59: An option to correct the segmentation.

At the end you can stop the segmentation or continue to the next file automatically by selecting yes in the last question on the screen (Figure 60).

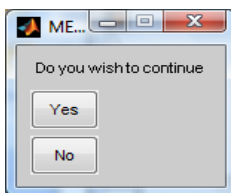


Figure 60: An option to stop or continue.

In case you want to stop the segmentation, pressing "No" would stop the program. After that the start and stop positions are written into an EXCEL file by the name: Subject1, or Subject2 ... depends on the subject that was selected. The EXCEL files are located in "MATLAB Segmentation" sub folder.

Appendix O - MATLAB Codes

Most of the work of this thesis was done using MATLAB, and thus it included a large number of functions and scripts. Several important scripts which were used for this thesis are shown below. The rest of the scripts are attached in the supplemented disc.

The functions names and explanation:

- CaliForces_Mini45 – Reads the ATI mini 45 force sensor
- MainMenu – A GUI function, which runs the segmentation process. A user manual is included in the disk supplements under the name 'readme'.
- LoadFilePhysical – Loads data from the physical experiment.
- Reconstruct – Reconstructs the tip end point using the three markers.
- Mat2Excel – Takes the data after several manipulations in MATLAB, and saves it to excel.

MainMenu:

This is the MainMenu Function. It includes a large set of commands relevant for the Gui construction but not for the thesis, and thus only the main part of the function is presented.

```
function varargout = MainMenu(varargin)
:
function Loadstart_Callback(hObject, eventdata, handles)
% hObject    handle to Loadstart (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc

%% Loading the Data and analyzing the data
stop=1;
handles.i=1;
handles.FlagLoad=0;

while stop==1 %we continue uploading files, unless the user decieds to stop
[handles]=Reconstruct(handles);
handles = TagID_New(handles.num1,handles.num2,handles);%
handles.trial = ReadHaptSphere_physi( handles.data,handles.startP);
[handles]=DosegFunc(handles.trial,handles);
Flag=1;
while Flag==1
    [handles]=InitDraw(handles);
    Flag=menu('Do you want to try again? ','yes','no');
end

guidata(hObject,handles);
stop=1;
stop = menu('Do you wish to continue','Yes','No');
ID=handles.ID
handles.i=handles.i+1;
Mat2Excel(handles); % Writes the data in Excel file
end
```

Mat2Excel

```
function handles=Mat2Excel( handles )
%MAT2EXCEL - this function writes the preprocessing data to an excel file
% input - Inside the handles there should be:
%         the entire name of the TXT file divided into:
%         "Name.sequence.num1.num2"
Sequence=str2double(handles.Sequence); % Changing to double format
num1=str2double(handles.num1);
num2=str2double(handles.num2);

TotalNum=(num1-1)*4+num2+(Sequence-1)*24+2; % Calculating the correct row in the excel
ExcelNum=['a' num2str(TotalNum)];
Name=handles.Name;
ExcelSheet=Name;

error=menu('do you have error ? ', 'early start', 'wrong movement', ' zero force' , 'no', 'Un Useable');
if error==2
    handles.ID2=inputdlg('what type?');
    handles.ID2=cell2mat(handles.ID2);
    handles.ID2=mat2str(handles.ID2);
end
if error==4
    error=0;
end
seg(1:7)=[Sequence num1 num2 error handles.st1 handles.en1 handles.Id];
if error~=2
    handles.ID2=handles.ID;
end
switch Name % changing the subjects names into numbers
    case 'Danping'
        Name='Subject1';
    case 'Mickelea'
        Name='Subject2';
    case 'Floerence'
        Name='Subject3';
    case 'Johanna'
        Name='Subject4';
end
xlswrite([Name, '.xls'], seg, ExcelSheet, ExcelNum);

seg1(1:2)={handles.ID handles.ID2};
ExcelNum=['h' num2str(TotalNum)];
xlswrite([Name, '.xls'], seg1, ExcelSheet, ExcelNum);

end
```


Reconstruct

The function for reconstructing the Tip from the three markers in the physical environment

```
function [handles]=Reconstruct(handles)
% load Tipbodyfinal
load Tipbody % Loading the reference point of the TIP
%%
[data handles] = LoadFilePhysical(handles); % Loading the data

handles.data=data;
[x,y,z, startP,startTip] = OrganizeData( data ); % separating the data for X, Y and Z
handles.x=x; handles.y=y; handles.z=z;handles.startP=startP;handles.startTip=startTip;
Len=length(x);
for i=startP:Len % The reconstruction process
    CenterX=mean(x(i,1:3)); CenterY=mean(y(i,1:3)); CenterZ=mean(z(i,1:3));
    O1x=CenterX; O1y=CenterY; O1z=CenterZ;
    % creating the three orientation vectors
    Ux=[x(i,2)-x(i,1) y(i,2)-y(i,1) z(i,2)-z(i,1)];
    Uy=[x(i,3)-x(i,1) y(i,3)-y(i,1) z(i,3)-z(i,1)];
    Uy=Uy/norm(Uy); Ux=Ux/norm(Ux); % Normalizing to Unit vector
    N=cross(Ux,Uy);
    Uz=N/norm(N);
    Uy=cross(Uz,Ux);
    Uy=Uy/norm(Uy);
    O=[O1x O1y O1z];
    U=[Ux' Uy' Uz' O'
        0 0 0 1];
    Tipbody(4)=1;
    Tcoda(i,1:4)=U*Tipbody;
end

% Re organizing the data so it would fit to older versions
handles.Tcoda=Tcoda/10;
handles.data(:,2:6)=handles.data(:,1:5);
a=16;
[Fx Fy Fz]=Cali_Forces(handles.data,a);
handles.Fx=Fx(startP:Len); handles.Fy=Fy(startP:Len); handles.Fz=Fz(startP:Len);
handles.data(:,3)=2;
handles.data(1:length(Tcoda),4:6)=Tcoda(:,1:3);
handles.data(1:length(Fx),7:9)=[Fx Fy Fz];

end
```

CaliForces_Mini45

The function for reading the data from the force sensor in the robotic experiment. For the force sensor in the physical experiment a different function was used, since it was not the same sensor.

```
function [forces]=CaliForces_Mini45
% Input – none
% Output – matrix of measured forces from the Mini 45 force sensor

persistent ai_mx    % Setting the Communication with the sensor

% Checking if it was already set
if isempty(ai_mx)
    ai_mx = analoginput('nidaq','Dev1');
    addchannel(ai_mx,0:5);
else
end

% The calibration matrix
CaliMat=[ 0.79508 -0.06630 -4.97248 31.84075 2.75546 -37.54836
          4.48963 -37.85949 -2.32447 18.42447 -2.39372 21.67771
          -18.57334 0.66213 -19.21617 0.36265 -18.62771 0.10539
          0.37628 -0.07474 34.78143 -0.65913 -34.09705 0.47792
          -38.16042 1.39559 19.99405 -0.45689 20.43803 -0.13513
          -1.98154 17.84681 -2.41389 17.31010 -1.75935 20.57352 ];

% The normalization factors
Norm=[0.359429723688808    0.359429723688808    0.144148748949902    16.0027504587186
      16.0027504587186 13.2208153624181];
%Normalizing the calibration matrix
CaliMat(:,1:6)=CaliMat(i,1:6)./Norm(i);
%Setting the frequency
frequency=1000;%Hz
time=0.01; % sec

set(ai_mx,'SamplesPerTrigger',frequency*time);
% Number of samples Per Trigger
set (ai_mx,'SampleRate',frequency);
start(ai_mx)
% trigger(ai_mx)
wait(ai_mx,20)
stop(ai_mx)
data=getdata(ai_mx); % Acquiring volts
forces=zeros(length(data),6); % Acquiring forces
forces(:,1:6)=CaliMat*(data(i,1:6));
end
```

LoadFilePhysical

The function for the automatic loading of files

```
function [data handles] = LoadFilePhysical(handles)
%LOADFILE
% Allows the user to load any file he wishes
% Output - Data file from the selected TXT and the analyzed file name
% The file should be in the following structure: Name.Sequence.num1.num2
% Sequence refers to the number of repetition, there were 5.
% num1 refers to the inner repetition there were 6.
% num2 refers to the movement number, there were 4.

if handles.FlagLoad==0; % checking if we already selected a subject?
[FileName,PathName] = uigetfile('*.txt','Select a file to check');
% path(path,PathName);
fid = fopen(FileName);
data=[];
filename=[PathName,FileName];
handles.filename=FileName;
handles.PathName=PathName;
file = importdata(filename);
data=file.data;
handles.FlagLoad=1;
else
% first updating the numbers of the new file experiment 2 physical
num2=str2double(handles.num2)+1;
handles.num2=['0',num2str(num2)];
if num2==5
    num1=str2double(handles.num1)+1;
    handles.num1=num2str(num1);
    handles.num2=['01'];
    if num1==7;
        handles.num1='1';
        Sequence=str2double(handles.Sequence)+1;
        handles.Sequence=num2str(Sequence);
    end
end
FileName=[handles.Name, ',',handles.Sequence, ',',handles.num1, ',',handles.num2, '.txt'];
filename=[handles.PathName,FileName];
handles.filename=FileName;
file = importdata(filename);
data=file.data;
end
%% Saving the file name and path

i=1;
while FileName(i)~='.' % Dividnig so that we can load more files
    Name(i)=FileName(i);
    i=i+1;
end
k=length(Name)+1;
i=i+1;
while FileName(i)~='.' % Dividnig so that we can load more files
    Sequence= FileName(i);
    i=i+1;
end
```

```
Len=length(Name)+length(Sequence)+2; % + 2 for all the '.'
num1=FileName(Len+1);

%Len=length(Name)+length(Sequence)+length(num1)+3; % + 3 for all the '.'
num2=FileName(Len+3:Len+4);
handles.Name=Name;
handles.Sequence=Sequence;
handles.num1=num1;
handles.num2=num2;

end
```


תקציר

תנועה מוגבלת היא תנועה אשר עבורה אחת (או יותר) מדרגות החופש הטבעיות של המערכת מוגבלת. תנועה מוגבלת הינה התנסות יומיומית וניתן למצוא אותה במגוון של פעילויות כגון: כתיבה, ציור ועוד. מחקרים קודמים שנעשו על תנועה מוגבלת התמקדו בעיקר במשטחים ישרים, אף אל פי כן חלק גדול מהמשטחים המגבילים הינם משטחים קעורים כגון: קערות, צלחות, כוסות ולכן יוצרים בעיה קינמטית ודינמית שונה. משטחים המיספריים הינם בעלי עקמומיות קבועה ויוצרים סביבת מחקר מעניינת עבור תנועה מוגבלת. אחד מהמכשולים העיקריים של חקר תנועה אנושית על משטחים שונים, הינו הצורך בבניה מחדש של הסביבה הניסויית עבור כל ניסוי. לכן סביבה וירטואלית שמאפשרת שינויי מבנה באופן פשוט יחסית, עשויה ליצור פלטפורמה מצויינת למחקר עתידי של תנועה אנושית באופן כללי. יחד עם זאת, סביבות וירטואליות צריכות תיקוף לפני שהן יתאימו למחקר תנועה. מחקר תנועה אנושית על משטחים עקומים מרחיב את הבנתנו לגבי תכנון ובקרת תנועה אנושית. בעתיד ניתן יהיה להשתמש בידע זה ליישומים שונים כגון פיתוח מדדים אובייקטיביים להערכת תנועה ורובוטים תומכי תנועה. מספר מחקרים בנושא תנועה חופשית (תנועות הצבעה) ותנועות ציור (על משטח ישר) מצאו שהגרביטציה השפיעה על המאפיינים הקינמטיים של התנועה. הבדלים נמצאו בפרופילי התאוצה והמהירות של תנועות אופקיות ואנכיות.

המחקר הזה כולל שלוש מטרות עיקריות: לתקף את המערכת הוירטואלית דרך השוואה לסביבה של פרופילי תנועה בסביבה הוירטואלית לפרופילי תנועה שנוצרו בסביבה פיסיקלית, לחקור את השפעות הגרביטציה על תנועה במשטח המיספרי ומימוש של אלגוריתמיקה לתנועה מוגבלת מתוך היפותיזה לגבי אסטרטגיות של תנועה אנושית על ידי מערכת רובוטית על מנת לקדם את ההבנה של תכנון ובקרה מכניים.

על מנת להשיג את המטרות הללו, שני ניסויים בוצעו, אחר בסביבה וירטואלית והשני בסביבה הפיסיקאלית שתוכננה להיות דומה ככל האפשר לסביבה הוירטואלית. הניסוי הוירטואלי כלל אחד עשרה נבדקים ומספר רב של תנועות למגוון רחב של כיוונים על ההמיספירה הוירטואלית. תנועות אופקיות ואנכיות נותחו לעומק על מנת לתקף את הסביבה הוירטואלי ועל מנת לאשש את ההשפעה של הגרביטציה על תכנון תנועה ובקרה של תנועה מוגבלת על ההמיספירה. בעקבות הידע שנצבר מהניסויים עם בני האדם, שלושה סוגים של בקרת הייבריד פותחו ונבחנו בניסוי רובוטי.

התוצאות הראו שישנם הבדלים בפרופילי התנועה בין הסביבה הפיסיקלית והוירטואלית. יחד עם זאת ההבדלים יכולים להיות מיוחסים להבדלים מבניים ופיסיקליים בין הסביבות ולא בהכרח להבדלים בתחושת ההיזון החוזר. לכן ההבדלים שנמצאו למעשה מחזקים את התיקוף של המערכת הוירטואלית.

מן התוצאות ניתן להסיק שהגרביטציה השפיעה על התנועה בשתי הסביבות. ההשפעה הייתה דומה למדי להשפעת גרביטציה שנמצאה במחקרים קודמים שנעשו על תנועה במשטח לא מוגבל (תנועות הצבעה) ובתנועות ציור על משטח ישר. לדוגמא זמן התאוצה מחולק לזמן התנועה השיג תוצאה גבוהה יותר בתנועה מלמעלה למטה, מאשר בתנועה מלמטה למעלה. עוד תוצאה מעניינת הייתה שתנועות אופקיות נטו כלפי מטה מתחת להטלה של קו ישר המחבר בין שתי נקודות המטרה, בעוד תנועות אנכיות התחלקו בצורה שווה בין שני צידי הקו הישר. תוצאות אלה מציעות שיתכן שהמוח לוקח את הכבידה בחשבון במהלך תכנון התנועה.

התוצאות של הניסוי הרובוטי מאששות שבקרת הייבריד הינה בקרה מתאימה עבור תנועה מוגבלת על המיספירה. עקב מגבלות חומרה התוצאות של הניסוי הרובוטי והאנושי לא היו ניתנות להשוואה איכותית, אך התוצאות היו מעודדות מכיוון שנמצאו מספר מאפיינים דומים. התנועה הרובוטית התאפיינה בהטלת תנועה ישרה (בדומה לניסוי האנושי) והתנועות האופקיות גם כן נטו כלפי מטה בעוד התנועות האנכיות לא. התוצאה האחרונה למעשה מביאה אותנו להציע שההשפעה של הגרביטציה על פרופילי התנועה הינה בחלקה תוצר של מאפיינים רובוטים הקשורים לגרביטציה.

מילות מפתח: תנועה מוגבלת, בקרת הייבריד, בקרת תנועה אנושית, סביבה וירטואלית


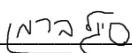
בחינה של בקרת תנועה מוגבלת על משטח המיספרי

חיבור זה מהווה חלק מהדרישות לקבלת תואר "מגיסטר" בהנדסה

מחבר: דני אייזיקוביץ

מנחה: ד"ר סיגל ברמן

אוניברסיטת בן-גוריון בנגב, הפקולטה למדעי ההנדסה
המחלקה להנדסת תעשייה וניהול

מחבר: דני אייזיקוביץ

מנחה: ד"ר סיגל ברמן

יושב ראש ועדת הוראה לתואר שני: פרופ' יוסף קרמר

דצמבר 2010

בחינה של בקרת תנועה מוגבלת על משטח המיספרי

חיבור זה מהווה חלק מהדרישות לקבלת תואר "מגיסטר" בהנדסה

מאת: דני אייזיקוביץ

מנחה: ד"ר סיגל ברמן

אוניברסיטת בן-גוריון בנגב, הפקולטה למדעי ההנדסה
המחלקה להנדסת תעשייה וניהול

דצמבר 2010