

## ABSTRACT

Title of the Thesis: DEVELOPMENT OF A NOVEL INTERACTIVE VISUAL TASK  
FOR A ROBOT-ASSISTED GAIT TRAINING IN STROKE

Amar Vamsi Krishna, Master of Science, 2018

Thesis Directed by: Professor Anindo Roy  
Department of Neurology  
University of Maryland School of Medicine

The goal of this thesis is to develop an interactive visual task for robot-assisted gait training after stroke. This is designed as a simple soccer-based computer video-game displayed on a screen, played by moving the ankle in dorsiflexion or plantarflexion to guide a soccer ball from its original position towards the goal. This stand-alone game is interfaced with the impedance controlled modular ankle exoskeleton (“Anklebot”) that provides assistance only as-needed, as an augmentative tool to further enhance ankle neuro-motor control and whole-body function after task-oriented robot-assisted treadmill walking. The design and features of the interactive video game, as well as the underlying biomechanical model that relates patient-to-game performance are presented. Simple adaptive performance algorithms are embedded, and bench tested to auto-adjust game parameters in real-time, concomitant to ongoing patient performance during robot-assisted therapy. Human in-loop testing strategies are proposed to validate the video-game performance and its feasibility for clinical use.

DEVELOPMENT OF A NOVEL INTERACTIVE VISUAL TASK FOR A ROBOT-  
ASSISTED GAIT TRAINING IN STROKE

by

Amar Vamsi Krishna

Thesis submitted to the Faculty of the Graduate School of the  
University of Maryland, College Park in partial fulfilment  
of the requirements for the degree of  
Master of Science  
2018

Advisory Committee:

Professor Anindo Roy, Chair  
Professor Mark Austin  
Professor Axel Krieger

©Copyright by

Amar Vamsi Krishna

2018

## PREFACE

The research conducted for this thesis forms a part of research collaboration led by Professor Anindo Roy at the Veterans Affairs Maryland Exercise and Robotics Center of Excellence (VA MERCE), Baltimore. The architecture and the source code in chapters 2 and 3 were developed by myself, with assistance from Suchitra Chandar. The research project, of which this thesis is a part, received Human Subject Research approval from the University of Maryland College Park (UMCP) Institutional Review Board (IRB), [1204238-1] Novel Interactive Visual Task for Robot-Assisted Gait Training in Stroke, Date 18<sup>th</sup> March 2018.

A version of the research presented in this thesis has been send for publication as A. Krishna, S. Chandar, R.S. Bama, and A. Roy, “Novel Interactive Visual Task for Robot-Assisted Gait Training for Stroke Rehabilitation”, seventh IEEE International Conference on Biomedical Robotics and Biomechatronics – Biorob 2018 and is currently under peer review.

This project would serve as a good introductory example to anyone interested to learn about systems engineering processes on a high level as the complete development lifecycle of a project starting from understanding the context and realizing a need for a new product (Chapter 1), needs analysis and requirements development, candidate architecture synthesis and response model development (Chapter 2), System realization/synthesis (Chapter 3), System Integration, Verification and Validation (Chapter 5).

## DEDICATION

In memory of my dear friend Ashwath Selvam. Your positive outlook towards life and curiosity inspires me.

## ACKNOWLEDGEMENTS

Firstly, I would like to express my deep gratitude to Professor Anindo Roy for mentoring me and giving me an opportunity to do my MS thesis within his group. This gave me the possibility of entering the exciting interdisciplinary field of Rehabilitation Robotics. Due to his helpful advice and continuous support, I was able to stay focused and not stray away from the essence of the work. His commitment and encouragement allowed me to submit my work at an International conference. My knowledge of Rehabilitation Robotics has greatly increased, thanks to his teachings in class and guidance throughout the course of this project.

I am grateful to my team mate Suchitra Chandar for her support and cooperation during the entire project. Her contagious work ethic is truly remarkable. My gratitude is also extended to Derek Eversley his help in demonstrating the standard procedures of a clinical trial and help in gathering the data for the study. Special thanks to Rahul S Bama for his insight and wonderful conversations during the course of this project.

I thank the members of my thesis committee, co-advisor Professor Mark Austin and Professor Axel Krieger for their pertinent deliberation and critique, helping me refine this document.

Sincere gratitude to Dr. John MacCarthy for teaching the art and science of Systems Engineering, both in class and outside. Lastly, I thank my peers for creating a challenging academic environment that accelerated the learning experience during the course

## Table of Contents

PREFACE.....	iv
DEDICATION.....	v
ACKNOWLEDGEMENTS.....	vi
Table of Contents.....	9
List of Figures and Tables.....	11
Chapter 1. Introduction.....	13
1.1. Conventional methods of Gait Training.....	13
1.2. Robotics in Stroke Rehabilitation.....	14
1.3. Design and Control of the Anklebot.....	16
1.4. Role of Ankle in Gait Biomechanics .....	18
1.5. Adaptive Ankle Robotics for Gait-Training.....	20
1.6. Clinical trials on Anklebot-assisted training.....	21
1.7. Use of Visual Tasks for Gait Training.....	22
1.8. Contributions of this thesis.....	25
Chapter 2. System overview and modeling.....	27
2.1. Requirements Development and analysis .....	27
2.1.1. Needs analysis.....	27
2.1.2. Operational concept .....	29
2.1.3. System Requirements development .....	40
2.2. Candidate system architecture .....	47
2.3. Mathematical response model of the system .....	49
Chapter 3. Game Development .....	55
3.1. Lifecycle model of the Software Development of the Video Game .....	55
3.2. System realization.....	56
Chapter 4. Interfacing and Verification.....	62
4.1. Hardware interfacing .....	62
4.2. Software interfacing.....	62
4.3. Verification.....	65
4.3.1. Unit Testing.....	65
4.3.2. Interface Testing - HRI Video Game Validation – “Causality” .....	65
4.3.3. Physical signal validation .....	66

4.3.4. Adaptive performance Validation.....	68
Chapter 5. Conclusions and Future work .....	70
References.....	72
Appendix -1: Requirements Traceability Matrix .....	79



## List of Figures and Tables

Fig 1: MIT's modular ankle robot system – Anklebot	17
Fig 2: Anatomy of a foot and ankle joint	19
Fig 3: Adaptive control of Anklebot timing	20
Fig 4: Conceptual model of human-robot-visual task interaction to promote two-way learning	25
Fig 5: Use case diagram of the video game system	30
Fig 6: Activity Diagram for performing seated isometric test	33
Fig 7: Activity Diagram for initializing the system	35
Fig 8: Activity Diagram from playing the game	37
Fig 9: Context Block Definition Diagram of the Anklebot system	38
Fig 10: System level Domain block definition diagram of the Video Game System	47
Fig 11: High level information flow diagram	47
Fig 12: System Level internal block diagram	48
Fig 13: Response model diagram of the system	49
Fig 14: Soccer game rudiments user interface window	50
Fig 15: Mathematical flowchart describing the adaptive performance algorithm	53
Fig 16: V Development Lifecycle model of the system	54
Fig 17: Conceptual flow diagram of netcat	63
Fig 18: Output characterization of the HRI – video game system	65
Fig 19: Bench testing of physical signal validation	66
Fig 20: Start line v/s time graph for an example trial for adaptive performance algorithm validation	67
Fig 21: Human torque v/s time graph for an example trial for adaptive performance algorithm validation	68

Table 1: User needs	29
Table 2: Use Case Narrative - perform seating isometric test	31
Table 3: Use Case Narrative – initialize game	34
Table 4: Use Case Narrative – play the video game	36
Table 5: Data dictionary	45
Table 6: Summary of spiral development	55

## **Chapter 1. Introduction**

Stroke is a leading cause of long-term disability, affecting nearly 800,000 people, with 75% being first-time incidences and a mortality rate of ~130,000 people in the United States every year [1]. The effects of stroke depend on several factors including the lesion location, resulting in various neurological complications with the most common one being contralateral hemiparesis [2]. Following hemiparesis, both the timing and magnitude of lower extremity (LE) muscle activity often undergoes radical changes due to impairment in both sensory and motor function [3,4]. For most survivors, this impairment causes a loss or impairment in ambulatory activity indexed by gait and balance dysfunction. This imposes higher energy demands in safe execution of activities of daily life (ADL), leading to a high risk of hip or wrist fractures caused by a fall, often in the first year alone [5-7]. Hemiparetic gait is accompanied by whole-leg (e.g. reduced paretic leg swing phase) and joint-specific (e.g. foot drop) biomechanical abnormalities. For example, a major complication resulting from drop foot is the slapping of the foot (“foot slap”) upon contact with the ground (as opposed to heel-first strike) and dragging of the toes during stance (“toe drag”). It is well established that motor rehabilitation in the form of physical therapy improves gait deficits post stroke due to neuro-plasticity [8,9] and motor learning [10].

### **1.1. Conventional methods of Gait Training**

Conventional gait training methods are targeted towards individual joint-impairment and concentrate on practicing different components of the gait like symmetric weight bearing and weight shifting, stepping training (swinging/clearance) and Push off/Calf rise training. Task specific gait training focuses on stimulating the weakened

muscle by repeatedly performing functional mobility tasks like walking and climbing. Yet, most stroke survivors live with residual deficits even after completion of conventional rehabilitative therapy, being forced to rely on assistive devices (ADs). In addition, PT-based gait training is labor intensive, often requiring one or more therapists per patient to manually assist with swing motion. Another gait training modality, treadmill aerobic exercise (TM), has proved to be more effective in improving gait speed and cardio-vascular fitness [11]. However, meta-analyses show that gait interventions, whether PT-mediated or TM-based exercise, produce only modest incremental gains in gait speed in those with mild-to-moderate gait deficits [12,13], enabling stroke patients to “limp” faster, but doing little or nothing to robustly improving underlying gait biomechanics or reduce dependence on ADs.

## **1.2. Robotics in Stroke Rehabilitation**

Ankle Foot Orthosis (AFO), a mechanical brace that rigidly locks up the ankle joint, is a conventional assistive technology for drop foot – a common impairment after stroke. Though the use of AFOs offer various biomechanical benefits by promoting a natural gait pattern, they are cannot be dynamically controlled. Hence there have been numerous studies to improve the rehabilitation post stroke. One of the interesting technologies developed is the use of computerized functional electrical stimulation, which can be either wearable (Eg WalkAide, Innovative Neutronics Inc., Austin, Tx and L300, Bioness Inc., Velencia, CA) or implantable (Bio Nerves or BIONs) to stimulate the deep peroneal nerve and tibialis anterior muscle to flex the ankle during swing. These technologies help in promoting better gait, but do not elicit neuroplasticity that can change the underlying biomechanical metrics of walking, thus rendering them dependant for a lifetime.

For over 6 decades now, research is being carried out in employing robotic exoskeletons and orthoses for motor rehabilitation. It is only in the past decade and a half that fully automatic electromechanical devices are being developed to account for the complex gait cycle biomechanics providing partial body support. Rehabilitation gait robots (modular or whole-leg) hold promise to promote lower limb motor recovery. Broadly, these devices fall under to categories:

- Wearable robotic exoskeletons that propel the paretic leg forward (e.g. Electromechanical Gait Trainer, GT1 [13,14], Lokomat [15], Robot Assisted Gait trainer (RAGT) [16])
- Actuated footplates attached under the sole of the foot that simulate different phases of gait (e.g. Haptic Walker [17]).

The GT1, was the first effort towards automating the gait-training process. It targeted the gait deficit of stance and swing phase imbalance by controlling the movement of the centre of mass. The patient was harnessed on two foot-plates on the treadmill that simulated the correct gait-cycle. The Lokomat is a robotic orthosis for the sole purpose of gait training through the use of a treadmill with body-weight support, aided by electro-mechanical drives. HapticWalker, like the GT1, consists of two foot-plates, but instead of focussing on gait balance, these plates mimic the full gait cycles. LOPES [18], was another popular lower extremity powered exoskeleton, consisting of one actuation at the knee-joint and two at the hip-joint and was constructed with the Virtual Model Control (VMC) architecture that calculates the actuation torques based on the simulated gait. Robot Assisted Gait Training (RAGT) using Active Leg Exoskeleton (ALEX) was developed for gait training for stroke survivors. It used a similar idea as Lokomat to assist the patient as needed by

incorporating a force-field controller at the knee and hip joints. MIT Skywalker [19] was a unique effort into gait rehabilitation and claimed to be the first orthotic device that takes into consideration the human dynamics and not just kinematics unlike most of the predecessors of its kind.

A radical change in rehabilitation robotics was introduced by MIT Manus [20,21], that incorporated impedance control [22,23,24] approach in its architecture. The MIT Manus is an upper-limb rehabilitation device. Clinical trials have shown a significant motor recovery and demonstrated patient safety. Current Lower Extremity (LE) robotics remains controversial, with consensus that current robot assisted approaches are inferior to usual care, or possibly deleterious [25,26]. Although these technologies are sophisticated, the operating principles are not aligned with contemporary motor neuroscience. In response, Roy et al have developed [27] and clinically tested [28,29,30] a modular, impedance-controlled, backdrivable, 2-Degree of Freedom (DOF) actuated ankle robot (“Anklebot”) to improve walking and balance functions after stroke, by means of increasing the paretic ankle contribution into task-oriented functional activities such as walking.

### **1.3. Design and Control of the Anklebot**

The unique features of the Anklebot mechanical design are its actuator back-drivability and intrinsically low mechanical impedance. The Anklebot is a wearable robot actuated in 2 Degrees of Freedom (DOF) and allowing free motion in all the 3 DOFs of the ankle joint. The actuation is provided in the sagittal and frontal planes or the dorsiflexion (DF)-plantarflexion (PF) and inversion-eversion (IE) of the ankle, via two linear actuators mounted in parallel as shown in Fig 1. If both actuators push or pull in the same direction, a DP torque is produced. If the actuators push or pull in relatively opposite direction,

inversion-eversion torque is produced. The Gruebler's mobility index of this exoskeleton is 3, which is the same as an ankle if modelled as a single joint.

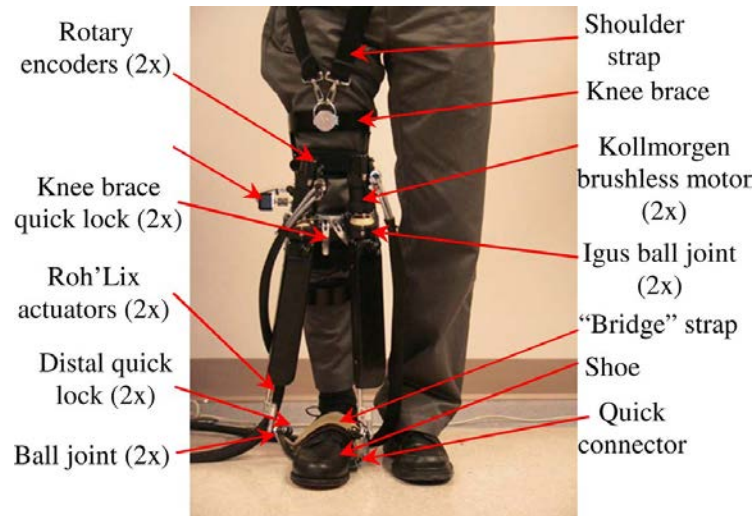


Fig 1. MIT's ankle robot system. Individual wearing the prototype Anklebot in standing position. Anklebot was designed to deliver therapy in seated overground, treadmill, and supine positions. Here, different components of the robot hardware are shown

This robot allows a free Range of Motion (ROM) in all the 3 DOF of the ankle (see section 1.4) - 25° of DF, 45° of PF, 25° of inversion, 20° of eversion, 15° of internal or external rotation, while providing actuation in DF-PF (sagittal plane) and IE (frontal plane) [27]. The Anklebot can deliver a continuous net torque of approximately 23Nm in DP and 15Nm in IE. It has low friction and inertia to maximize back-drivability.

The impedance controlled Anklebot provides task-oriented locomotor training after stroke. It is a clinically proven gait training tool by increasing the contribution of the paretic ankle in walking function. A novel gait event-triggered, sub task algorithm enables precise timing of robotic assistance. The gait training involves an adaptive approach in that, the training parameters are incremented progressing gradually towards a more natural gait subject to the performance and tolerance of the stroke patient. The focus was on the ankle

alone unlike most of the gait trainers as studies [31, 32] have shown that ankle impairments have the greatest effect on gait balance and to some extent the gait velocity.

#### **1.4. Role of Ankle in Gait Biomechanics**

The ankle joint complex is comprised of the lower leg and the foot and forms the kinetic linkage allowing the lower limb to interact with the ground, a key requirement for gait and other activities of daily living. Despite bearing high compressive and shear forces during gait, the ankle's bony and ligamentous structure enables it to function with a high degree of stability and compared with other joints such as the hip or knee, it appears far less susceptible to degenerative processes such as osteoarthritis, unless associated with prior trauma [33].

The “foot and ankle” is made up of the twenty-six individual bones of the foot, together with the long-bones of the lower limb to form a total of thirty-three joints as shown in Fig 2. Although frequently referred to as the ‘ankle joint’, there are a number of articulations which facilitate motion of the foot. The ankle joint complex is made up of the talocalcaneal (subtalar), tibiotalar (talocrural) and transverse-tarsal (talocalcaneonavicular) joint.



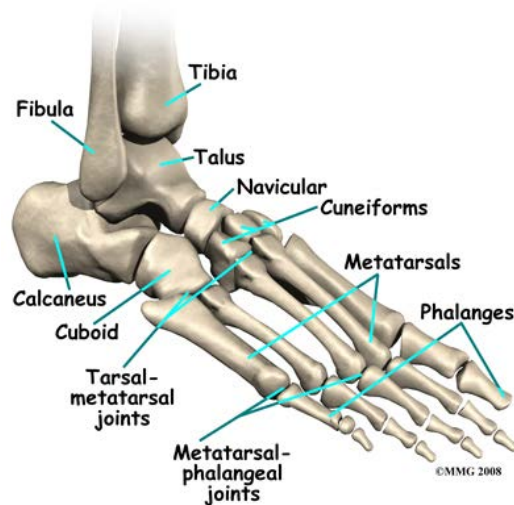


Figure 2. Anatomy of the foot and ankle joint. Major bones that make up this joint are shown in the figure. Image source: <https://midwestbonejoint.com/child-orthopedics/congenital-flatfoot-pes-planus/>

The key movement of the ankle joint complex are PF and DF, occurring in the sagittal plane; internal and external rotation occurring in the transverse plane and IE, occurring in the frontal plane. Combinations of these motions across both the subtalar and tibiotalar joints create three-dimensional motions called supination and pronation. Both terms define the position of the plantar surface of the foot (sole). During supination, a combination of plantarflexion, inversion and adduction causes the sole to face medially. In pronation, dorsiflexion, eversion and abduction act to position the sole facing laterally.

The ankle ROM has been shown to vary significantly between individuals due to geographical and cultural differences based on their activities of daily living, in addition to the method used for assessing ROM [34]. Motion of the ankle occurs primarily in the sagittal plane, with plantar- and dorsiflexion occurring predominantly at the tibiotalar joint. Several studies [34,35] have indicated an overall ROM in the sagittal plane of between 65 and 75°, moving from 10 to 20° of dorsiflexion through to 40–55° of plantarflexion. The total range of motion in the frontal plane is approximately 35° (23° inversion – 12°

eversion). However, in everyday activities, the ROM required in the sagittal plane is much reduced, with a maximum of 30° for walking, and 37° and 56° for ascending and descending stairs, respectively [36].

### 1.5. Adaptive Ankle Robotics for Gait-Training

A novel adaptive control algorithm was used to identify the sub events of gait and precisely time the assistance [37]. This customizes robotic therapy to individual gait deficits (e.g., foot drop, weak push-off) and precisely timed impedance control support, in turn affording safety and maximum autonomy to effectively integrate the Anklebot into the context of locomotor learning as shown in Fig 3. A novel feature of the adaptive controller is the use of robust systems control that tolerates perturbations due to step-to-step variability, thereby increasing operational stability, and accommodates heterogeneous levels of mobility across the spectrum of stroke recovery. The underlying concept is to precisely time robotic assistance, if needed, to gait sub-events derived from real-time signals via bilateral micro-switch insoles.

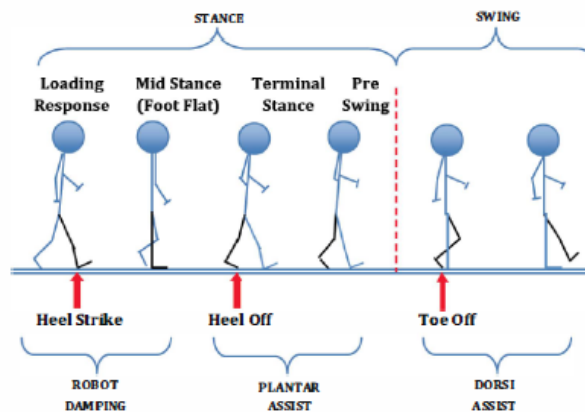


Figure 3. Underlying principle of adaptive Anklebot timing: the robot delivers assistance during key gait sub-events

This breakthrough allows, for the first time, the ability to:

- Differentially target deficits in both stance and swing phases of gait, enabling tailoring of robotic assistance to individual deficits (e.g., foot drop, weak push-off, improper landing) via gait sub-event triggered actuation
- Tolerate step-to-step variability to prevent destabilization and ensure patient safety
- Progress and dynamically modulate robotic outputs, both in the immediate time frame (step-to-step) and over the course of therapy (inter-visit), as subject performance adapts.

This novel *co-robotics* application defines a cooperative learning process between the subject and robot, which over time dynamically shapes robotic outputs to promote and elicit greater volitional effort toward durable locomotor learning. Hence, adaptive control co-robotics holds promise to shift practice paradigms for the care of hemiparetic stroke, providing a bioengineering solution to a previously immutable neurological deficit.

## **1.6. Clinical trials on Anklebot-assisted training**

Numerous clinical studies have shown that the Anklebot can improve paretic ankle motor control while reducing paretic ankle impairments, which translate to whole-body improvements like 20% higher floor walking speeds [28,29]. In the most recent clinical study [30], the Anklebot was integrated into treadmill training (TMR) using an adaptive control to precisely time robotic activation to sub-events and ankle directionality across the gait cycle [37]. This approach enabled the customization of robotic therapy to individual gait deficits (e.g., foot drop, weak push-off) and precisely timed impedance control support, in turn affording safety and maximum autonomy during robot-assisted walking practice

[37]. The primary finding was that six weeks of TMR (3×weekly) durably improves gait biomechanics and paretic ankle function during independent walking in chronic stroke survivors [30]. Underlying these improvements were a progressive increase in unassisted paretic swing to normal levels with retained improvements 6 weeks after cessation of training, which caused a majority of TMR graduates (85%) self-discard their orthotics or reduce reliance on their ADs. In addition, TMR significantly increased stance propulsive impulses to near-normal levels at retention, which contributed toward ongoing increases in gait velocity after training ended. This is the first therapy, robotic or otherwise, to therapeutically improve functional dorsiflexion and restore impaired push-off during independent walking in chronic stroke enabling individuals to more ambulatory in home and community settings, by means of greater and better engagement of their paretic ankle.

### **1.7. Use of Visual Tasks for Gait Training**

Patient nonadherence in therapy is a major barrier to rehabilitation. Recovery is often limited and requires prolonged, intensive rehabilitation that is time-consuming, expensive, and difficult. Research suggests that video games are beneficial for cognitive and motor skill learning in both rehabilitation science and experimental studies with healthy subjects. Physiological data suggest that game play can induce neuro-plastic reorganization that leads to long-term retention and transfer of skill [8-10]. However, more clinical research in this area is needed. There is interdisciplinary evidence suggesting that key factors in game design, including choice, reward, and goals, lead to increased motivation and engagement [38].

Various studies [38-41] have been conducted to evaluate the efficacy of incorporating a visual task to help in the process of neurorehabilitation –robot-assisted or otherwise. For example, treatments such as robot-assisted repetitive task practice (RTP) and virtual rehabilitation activities [42-47], have been shown to produce improvements in hand function, but have yet to reinstate function to pre-stroke levels—which likely depends on developing the therapies to impact cortical reorganization in a manner that favours or supports recovery.

A fundamental conceptual question in stroke neuromotor rehabilitation is whether to emphasize task specific gait pattern training, or modular and joint specific mass training aimed at specific stroke impairments. A 6-week (3×weekly = 18 sessions) seated performance-based Anklebot training to determine initial feasibility for using the ankle robot in extended training in individuals with chronic and sub-acute stroke [28,29] has shown significant results. A performance-based training protocol was implemented with subjects playing a “racer” videogame, adapted from the MIT Manus [20,21] protocols, that requires repetitive DF and PF of the paretic ankle to move a screen cursor “up or down” in order to pass through “gates” that approached across the screen at different vertical levels. Gate locations were individualized for each subject based on paretic ankle ROM, and level of assistance was set initially to facilitate an 80% success rate, while the level of robotic support was reduced every 2 blocks (160 movements), from 125—75—25 Nm/rad, increasing the volitional movement demands on the paretic ankle. Sessions also included unassisted trials before and after training, bringing the total targeted movements to 560 per day to fit within a one hour training session, including rest intervals between blocks of trials. The primary finding was that this visually-evoked, visually-guided seated Anklebot

training significantly improved all robotics measured parameters of paretic ankle motor control and selected spatiotemporal gait parameters in chronic hemiparetic patients that had already completed all conventional rehabilitation options.

While the seated visuomotor approach with the Anklebot showed promise as a motor learning platform for persons with chronic hemiparetic stroke, it represented only the first attempt at gauging the device's feasibility for use across the phases of generally. A recent clinical study incorporating an adaptive control approach for Anklebot mediated motor rehabilitation of the whole-body through a task-oriented treadmill-based gait training post stroke has shown a greater impact [30]. Moreover, the positive effects of motivation and bio-feedback through visual task integration into robotics has been demonstrated as the subjects who trained with the interactive video game using the Anklebot were found to have more efficient cortical dynamics i.e., reduced networking [48].

Few prior studies including the visually guided seated Anklebot training, have used sensory-motor enhancements like visual tasks as a cueing and motivational stimulus, to augment robot-mediated functional outcomes. And even bigger gap is in “closing the learning loop” that is, to implement co-operative learning in which the robot's performance is continually or periodically updated based on ongoing (step-by-step) human performance which is shown in Fig. 4.

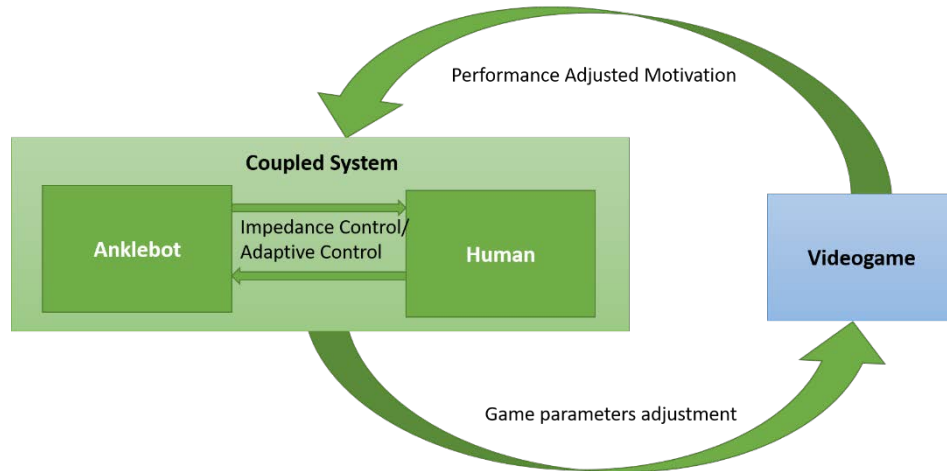


Figure 4. Conceptual model of human-robot-visual task interaction to promote two-way cooperative learning.

### 1.8. Contributions of this thesis

The gap of “closing the learning loop” described in section 1.7 forms the foundation of this research. The overarching *hypothesis* is “An interactive visual task when integrated with treadmill training, will further increase the ankle neuro-motor and whole-body functional gains and learning rates, and that these performance changes can shape robotic outputs to close the learning loop”. The scope of this project is to develop a comprehensive stand-alone platform, that can be easily integrated with the existing Anklebot assisted gait training protocol to help test this hypothesis in a clinical setting with actual stroke patients. Hence, the contributions of this thesis are as follows:

- a) Understanding the needs of the user, concept analysis and requirements development
- b) Development of a conceptual mathematical model for a novel interactive visual task that integrates with the current treadmill-training protocol
- c) Development of a simple adaptive performance algorithm to create a framework enabling “two-way (co-operative) learning”

- d) Design and development of a prototype of a software system derived from the conceptual model
- e) Development of a hardware-software framework for stand-alone visual tasks for easy interfacing with an existing rehabilitative robotic system
- f) Verification of the software package and its interfacing with the existing Anklebot system – both hardware and software, and validation of the conceptual model without a human in the loop

The end-product from this research would be a software add-on platform that is ready for a usability test involving healthy patients. Post a successful completion of a usability test, this software package would be integrated in a clinical test involving stroke patients undergoing a robot assisted gait therapy, the results of which would be compared against a similar clinical test without a visual task integration, which would help in testing the overarching hypothesis.

A summary of this research has been submitted to the seventh IEEE International Conference on Biomedical Robotics and Biomechatronics – Biorob 2018 (Novel Interactive Visual Task for Robot-Assisted Gait Training for Stroke Rehabilitation, Krishna, Chandar, Bama, and Roy), and is currently under peer review.



## **Chapter 2. System overview and modeling**

The system of interest here is a motivational visual task in form of a video game which shall be referred to as the Video Game (VG) system from hereon. The system externally interacts directly or indirectly with hardware-software elements of the Anklebot System along with a direct interaction with the human as described in section 1.7. This interdependency warrants the application of Systems Engineering(SE) concepts and processes especially Model Based Systems Engineering (MBSE) processes and concepts. Informal application of Systems Engineering – use of some SE concepts, processes and tools have been used to develop the system. Most systems engineers agree to the following five systems engineering processes that are most commonly iterated:

- a. Requirements Development and analysis
- b. Architecture Development and analysis
- c. Design Development and analysis
- d. Implementation
- e. Verification and/or validation and analysis

This research project aims to incorporate all the five major SE processes in its development lifecycle as described in the following sections.

### **2.1. Requirements Development and analysis**

#### **2.1.1. Needs analysis**

The primary objective of the needs analysis phase of the system life cycle is to show clearly and convincingly that a valid operational need (or potential market) exists for a new system or a major upgrade to an existing system, and that there is a feasible

approach to fulfilling the need at an affordable cost and within an acceptable level of risk [49]. Based on the current research in rehabilitation robotics described in chapter 1, there clearly exists a need for the development of a stand-alone visual task in form of an interactive real time video game that can provide performance adjusted motivation. The Motivational Visual Task (MVT) for ankle rehabilitation must satisfy clinical and biomechanical requirements necessary to promote volitional effort and thereby elicit neuroplasticity toward durable locomotor learning along with the associated programmatic and human factors requirements. To have a therapeutic significance, the visual task for ankle rehabilitation- the Video Game (VG) System functionally must be linked directly or indirectly to walking, and should also be easily relatable to the geriatric, stroke patients. In addition to providing motivation to the patients through bio-feedback, the visual interface must capture relevant Human-robot interaction (HRI) metric i.e. human vs. robot torque contribution. This is a foundational necessity for embedding a two-way adaptive learning algorithm within the VG system which “closes the learning loop”, as described in section 1.7. The primary requirement is to develop game environment and cursor control such that cursor movement is determined by clinically relevant kinetic or kinematic metric that can be either measured or be derived from the robot (e.g. peak swing angle, maximum human torque). User needs were derived from interacting and closely monitoring geriatric stroke patients, and physical exercise therapists – the primary stakeholders and the end user of this system. The user needs are as listed in Table 1. These user needs would be useful in the final system validation (beyond the scope of this research). Formal system level and in turn element level requirements (functional and operational) would be

derived from these user needs. Note that some of the user needs are subjective and ambiguous. The developed system requirements should confirm with the user needs described below within the bounds of the applied constraints (discussed in section 2.1.3).

<b>Identifier Number</b>	<b>User Needs</b>
UN1	MVT shall provide bio feedback for robot-aided neurorehabilitation of walking
UN2	MVT shall be easily relatable to elderly stroke patients
UN3	MVT shall be linked to human gait metrics
UN4	MVT shall provide quantitative bio-feedback through an incremental scoring system
UN5	MVT shall capture real time Human Robot Interaction metric(s) of gait
UN6	MVT shall have a dynamic display – real time movements on the screen
UN7	MVT shall be interactive – real time performance adjusted
UN8	MVT shall be an augmented task – add on to the preexisting gait training protocol
UN9	MVT shall allow user to select between dorsiflexion and plantarflexion assist

Table -1: User needs derived from Stakeholders.

### **2.1.2. Operational concept**

The user needs described in the previous section are to be fulfilled by integration of the MVT with the existing Anklebot mediated motor rehabilitation of the whole-body through a task-oriented treadmill-based gait training post stroke protocol [30]. The principal objective of operational studies is the definition of the objectives, in operational terms, that a new system must meet in order to justify its development [50]. In this needs-driven development, these objectives must overcome the gaps in the

current system, described in section 1.7 that the overarching hypothesis testing of this research is trying to test. The term “objectives” is used in place of “requirements” because at this early stage of system definition, the latter term is inappropriate. It should be anticipated that many iterations would take place before the balance between operational performance and technical risk, cost, human factors, and other developmental factors will be finally established. Although objectives should be quantifiable and objective, most are subjective and qualitative at this stage as seen in section 2.1. In order to quantify these objectives and refine them into verifiable system requirements, focus should be on the purpose of the system in a large sense. A SysML Use Case Diagram (UCD) is a very useful tool to establish the operational behaviour of the system within its environment. The UCD of the Video Game (VG) system that would provide visual bio-feedback to stroke patients undergoing robot mediated gait therapy is as shown in Fig 5.

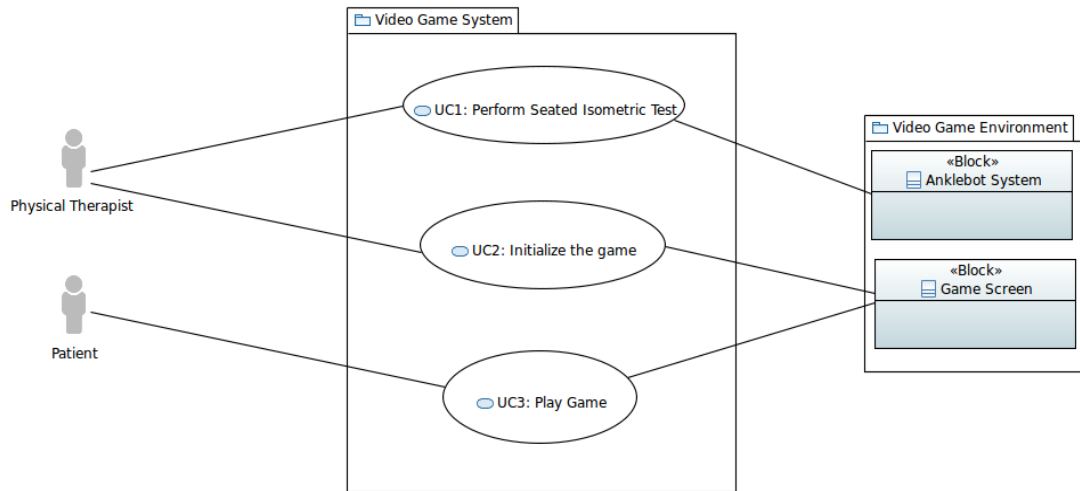


Figure 5. A Use Case Diagram of the VG System Describing the Activities performed during the standard operational Robot Assisted treadmill gait therapy session in conjunction with the video game

A use case diagram concisely conveys a set of use cases—the externally visible services that a system provides—as well as the actors that invoke and participate in those use cases. A use case diagram is a blackbox view of the system; it is therefore well suited to serve as a system context diagram. A use case captures a contract between the stakeholders of a system about its behavior. The use case describes the system’s behavior under various conditions as it responds to a request from the physical therapist -our primary actor. The VG system responds, while the stroke patient is interacting with it, protecting his/her interests. Different sequences of behavior, or scenarios, unfold, as shown in the diagram. The use case gathers those different scenarios together. In the actual deployment of the video game system as an add-on to the Anklebot assisted gait training therapy, certain operational procedures need to be performed during the training protocol. Table 2 describes the scenario of performing a seated isometric test which helps the Physical Therapist (PT) tune the impedance controller of the Anklebot System. The Anklebot has programmable, user defined controller gains  $K$  and  $B$ , that help estimate the paretic ankle stiffness -which is a critical component of the underlying impedance controller architecture. It functionally estimates the ankle stiffness and damping and provides ‘assist as needed’ for the selected training protocol. A seated isometric is performed to experimentally estimate the ankle stiffness and damping. It can also be used to measure the maximum torque that can be generated by the paretic ankle during seated dorsiflexion or plantarflexion.

Title	UC1: Perform Seated Isometric Test
Date	20 <sup>th</sup> April 2018

Version	1.1
Summary	Perform Seated Isometric Test
Actor/User	Physical Therapist (PT), Patient
Preconditions	<ul style="list-style-type: none"> <li>• The patient is seated</li> <li>• The Anklebot is correctly donned on the patient</li> <li>• Anklebot hardware is functioning correctly</li> <li>• Anklebot is calibrated correctly</li> </ul>
Trigger	The PT determines they wish to perform a seated isometric test on the stroke patient
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The PT opens the Anklebot scales adjustment window</li> <li>2. The PT sets the spring constant 'K' and damping coefficient 'B' to 0</li> <li>3. The PT asks the patient to exert maximum force to push the foot downward</li> <li>4. If the patient can move the foot downwards, the PT slightly increases the K and B values within given stability range</li> <li>5. Repeat step 3 and 4 until the patient cannot push the foot further down.</li> <li>6. Note down the B, K and the Torque values</li> <li>7. Repeat steps 2 through 6 for pulling the foot upwards</li> <li>8. End</li> </ol>
Post Condition	The patient is ready to start the Robot assisted gait therapy integrated with the Video Game for motivation
Alternate Scenario 1	
Trigger	The Anklebot scales adjustment window does not open
	<ol style="list-style-type: none"> <li>1. The PT checks if all the three lights on the Anklebot computer are in on state</li> <li>2. If not, PT restarts the Anklebot software system</li> <li>3. PT returns to step 1 of the main success scenario</li> </ol>
Alternate Scenario 2	
Trigger	The Anklebot System fails to provide resistance to foot movement

	<ol style="list-style-type: none"> <li>1. PT removes the Anklebot from the patient's leg</li> <li>2. PT performs hardware check and notifies the lab manager</li> </ol>
Failure Scenario	
Trigger	The patient is not able to push or pull the foot even at $K=0$ and $B=0$
Post condition	The patient is switched from robot assisted gait training to seated robot training protocol

Table 2: Use Case Narrative for UC1 – Perform Seated Isometric Test

The activity diagram associated with UC1 is shown in Fig 6.

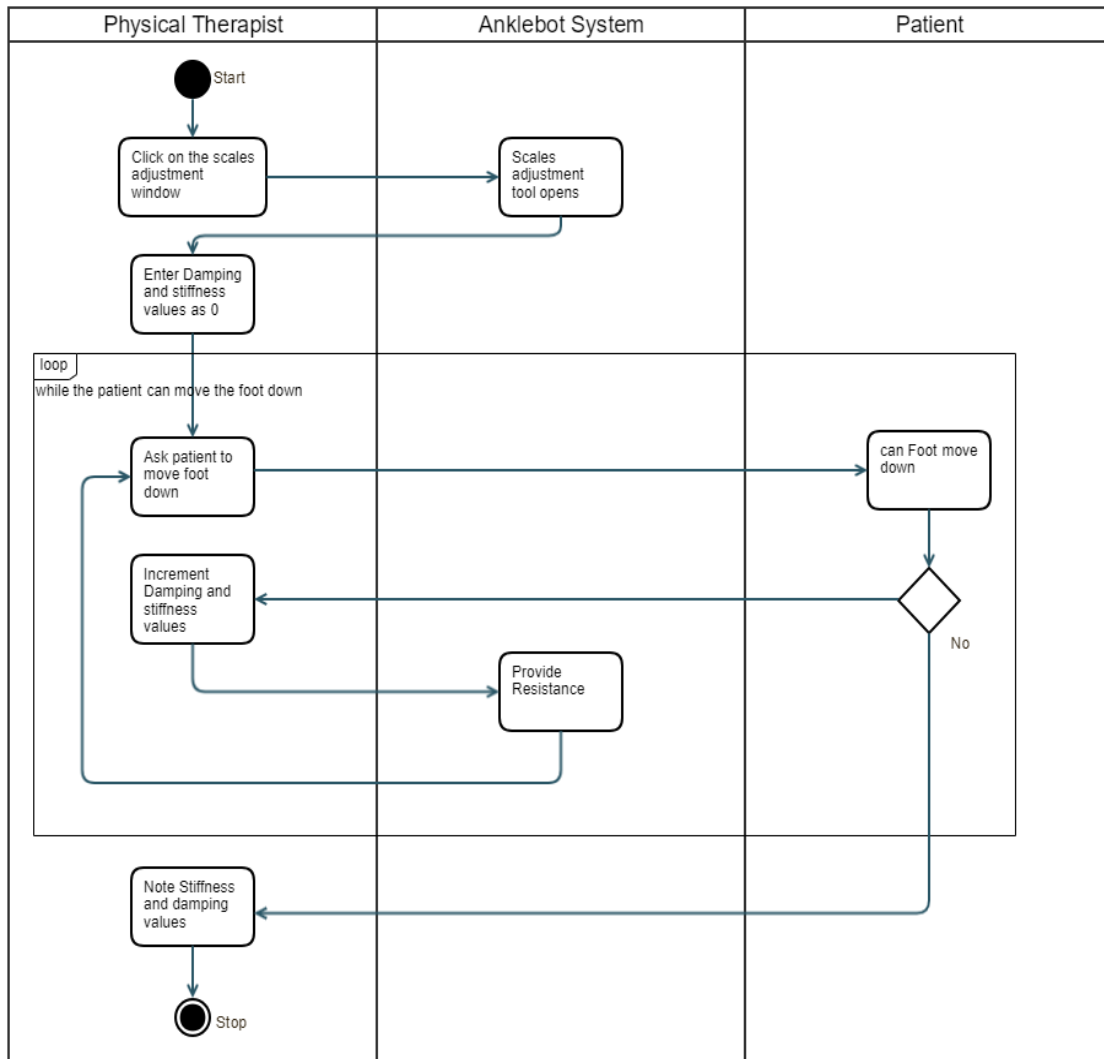


Figure 6. Activity Diagram for UC1 – Perform Seated Isometric Test

Use case-2 (indexed as UC2) describes the calibration procedure performed by the physical therapist to set up the correct game parameters. The mathematical relations between the user input factors and their relevance to physical therapy and the video game is described in section 2.3. A use-case narrative for UC2 describing the sequence of activities performed is described in Table 3.



Title	UC2: Initialize the game
Date	20 <sup>th</sup> April 2018
Version	1.1
Summary	Initialize the game
Actor/User	Physical Therapist (PT)
Preconditions	<ul style="list-style-type: none"> <li>• Seated isometric test has been successfully performed</li> <li>• The external game screen has been successfully interfaced with the Anklebot computer</li> <li>• The VG system is running bug free</li> </ul>
Trigger	The PT determines they wish to initialize the game
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The PT opens the Video Game program GUI – user input menu</li> <li>2. PT selects the mode of operation – dorsiflexion assist/plantarflexion assist</li> <li>3. PT enters the values of threshold torque and pixel offset, <math>\beta</math></li> <li>4. PT calculates the torque-to-screen pixels sensitivity coefficient, <math>\alpha_{calib}</math> using eq (7) in section 2.6</li> <li>5. PT enters the value of <math>\alpha_{calib}</math> in the user input menu GUI</li> <li>6. PT selects the type of adaptive performance metrics and the associated window size</li> <li>7. PT hits Go to Game</li> <li>8. End</li> </ol>
Post Condition	Game window with moving graphics pops up on the screen
Alternate Scenario 1	
Trigger	Moving game element not seen
	<ol style="list-style-type: none"> <li>1. PT stops the game</li> <li>2. PT verifies the calculations of <math>\alpha_{calib}</math></li> <li>3. PT returns to step 1 of the main success scenario</li> </ol>

Table 3: Use Case Narrative for UC2 – Initialize game

The associated activity diagram for the UC2 is described in Fig 7.

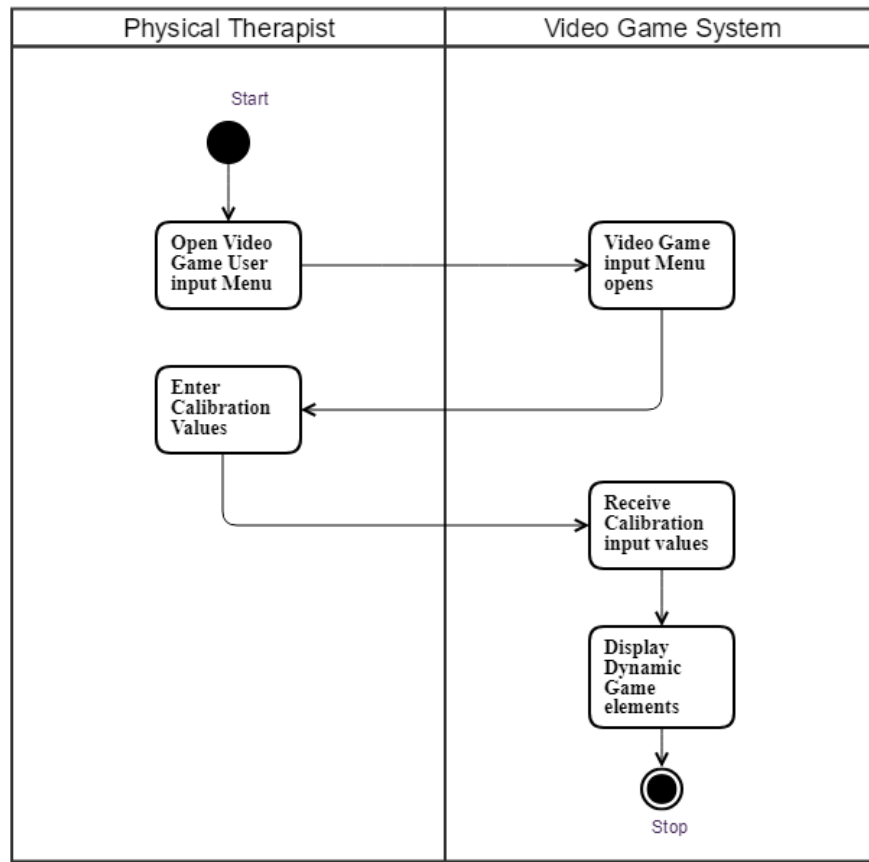


Figure 7. Activity Diagram for UC2 – Initialize the system

Use case-3 (indexed as UC3) describes the actual assistance procedure of the Anklebot assisted gait training with an integrated video game as a motivational visual task. A use-case narrative for UC3 describing the sequence of activities performed is described in Table 4

Title	UC3: Play game
Date	20 <sup>th</sup> April 2018
Version	1.1
Summary	Play the Video Game
Actor/User	Patient, Physical Therapist (PT)
Preconditions	<ul style="list-style-type: none"> <li>The Anklebot gait training program is running bug free</li> </ul>

	<ul style="list-style-type: none"> <li>• Patient is harnessed</li> <li>• Patient is standing of the treadmill</li> </ul>
Trigger	The PT starts the Treadmill based robot assisted gait training
Main Success Scenario	<ol style="list-style-type: none"> <li>1. The PT selects the treadmill speed</li> <li>2. The patient walks on the treadmill</li> <li>3. Patient plays the real time game on a game screen through walking</li> <li>4. Repeat step 3 till the end of trial time</li> <li>5. PT stops the treadmill</li> <li>6. PT ends the game</li> <li>7. PT ends the game protocol</li> <li>8. PT removes the harness from the patient</li> <li>9. Patient is seated</li> </ol>
Post Condition	The patient performance is stored in the database of the Anklebot system
Alternate scenario	<ol style="list-style-type: none"> <li>1. Patient gets fatigued before the end of trial</li> <li>2. PT stops the robot assisted TMR training and the game</li> <li>3. End</li> </ol>
Trigger	Patient gets fatigued before the end of trial
	PT performs steps 5-9 of the main success scenario

Table 4: Use Case Narrative for UC3 – Play the video game

The associated activity diagram for the UC2 is described in Fig 7.

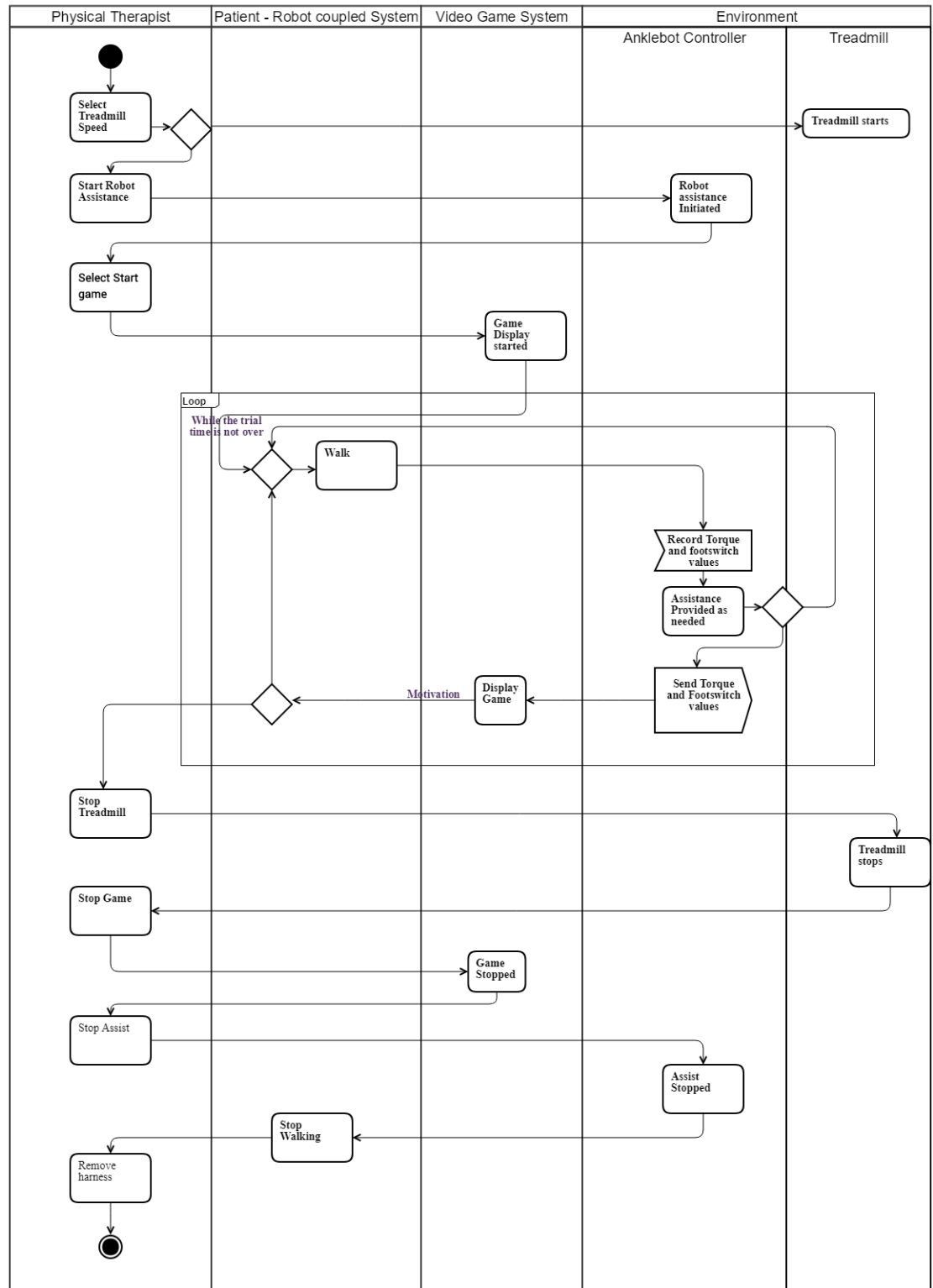


Figure 8. Activity Diagram for UC3 – Play the video game

A SysML Block Definition Diagram (BDD) for the Anklebot System is shown in Fig 9 to help explain the structural and functional relevance of the VG system as a part of the Anklebot System.

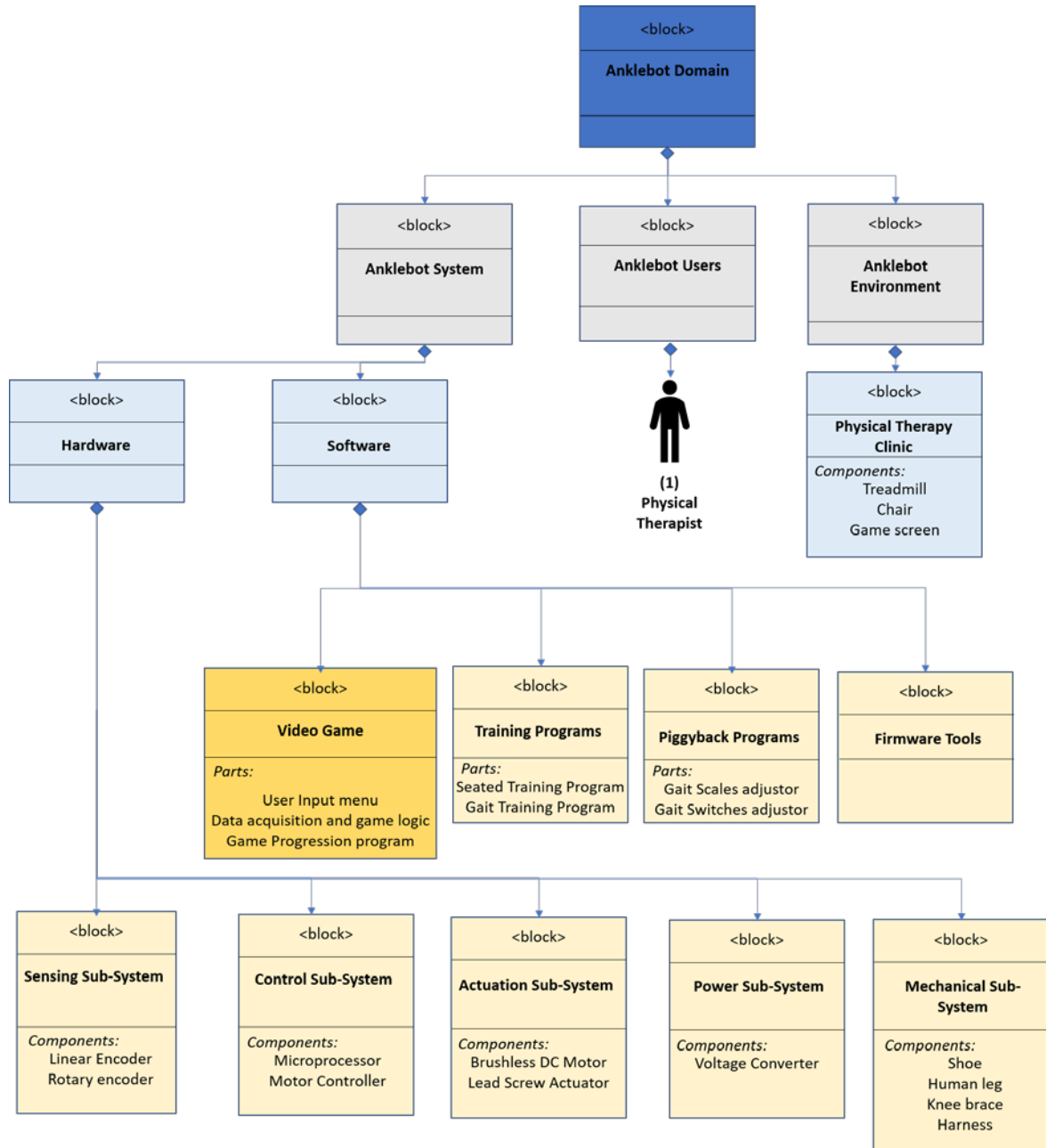


Figure 9. Context Block Definition Diagram of the Anklebot System

### **2.1.3. System Requirements development**

Sections 2.1.1 and 2.1.2 discussed the process of needs and operational analyses to provide a well-documented justification for the development of a motivational visual task in the form of a video game to be used in conjunction with the Anklebot assisted treadmill-based gait training protocol to rehabilitate neuromotor functions of a patient post stroke. In this section, the operationally oriented system objectives are converted into an engineering-oriented view by formally describing the verifiable system requirements. Other than satisfying the underlying bio-mechanical and clinical needs described in section 2.1.1, the system should satisfy certain programmatic and human factor needs subject to appropriate constraints. Based on the Stakeholder needs, a system of interest developed is a novel and simple interactive visual task, that is packaged as a single input (robot torque)—output (screen cursor) computer Video Game (VG System). It is based on the penalty kick scenario in a soccer game. The objective is to provide step-by-step bio-feedback during assisted-walking to facilitate patient's walking performance (e.g. swing DF clearance). A key feature of the game is that it challenges the patient based on their performance by incorporating an adaptive learning algorithm. Since this research is an academic project, the schedule constraint is to have a minimum viable product – a prototype of a game satisfying all of the listed requirements before the graduation deadlines set by the Graduate school of University of Maryland, College Park. Additionally, this project is predominantly is predominantly conceptual and abstract modeling and software development and integration using either open source tools or the ones made freely available for the

students of University of Maryland, College Park. Hence, cost as a constraint has not been addressed in the requirements document. The comprehensive list of requirements derived are as follows:

SR1. System modes of operation

SR1.1. The VG System shall have a Dorsiflexion assist mode of operation

SR1.1.1. The VG system shall have a Dorsiflexion mode of operation without an adaptive performance change mode

SR1.1.2. The VG system shall have a Dorsiflexion mode of operation with an adaptive performance change mode

SR1.1.2.1. The VG system shall have a Dorsiflexion mode of operation with an adaptive performance change mode based on the average of the peak human torques within a user defined window size

SR1.1.2.2. The VG system shall have a Dorsiflexion mode of operation with an adaptive performance change mode based on the correlation coefficient of peak human torques and time within a user defined window size

SR1.2. The VG System shall have a Plantarflexion assist mode of operation

SR1.2.1. The VG system shall have a Plantarflexion mode of operation without an adaptive performance change mode

SR1.2.2. The VG system shall have a Plantarflexion mode of operation with an adaptive performance change mode

SR1.2.2.1. The VG system shall have a Plantarflexion mode of operation with an adaptive performance change mode based on the average of the peak human torques within a user defined window size

SR1.2.2.2. The VG system shall have a Plantarflexion mode of operation with an adaptive performance change mode based on the correlation coefficient of peak human torques and time within a user defined window size

## SR2. System capability requirements

SR2.1. The VG system shall be event triggered

SR2.1.1. The VG system shall record the discrete footswitch voltages to identify the event trigger

SR2.2. The VG system shall correctly calculate the current position of the ball on the screen for a given torque value

SR2.2.1. The VG system shall ensure the ball never moves away from the goal post during the trial

SR2.3. The VG system shall correctly calculate the performance increase or decrease in a given window of trials



SR2.4. The VG system shall correctly calculate the positional change of the starting position of the ball corresponding to the relative increase or decrease in the performance

SR2.4.1. The VG system shall evaluate the increase or decrease in the performance as new start distance of the ball

SR2.5. The VG system shall have an incremental scoring system based on the final position of the ball at the end of a gait cycle

### SR3. System interface requirements

SR3.1. External interface requirements

SR3.1.1. External hardware interface requirements

SR3.1.1.1. The external hardware interface of the VG system with the environment shall be an RJ45 ethernet cable

SR3.1.2. External software interface requirements

SR3.1.2.1. VG System input requirements

SR3.1.2.1.1. The VG system shall accept real time discrete foot switch values from the environment

SR3.1.2.1.2. The VG system shall accept real time torque values from the environment

SR3.1.2.1.3. The VG system shall accept input from the user through the user input menu

SR3.1.2.1.3.1. The VG system shall accept the torque-to-screen pixel sensitivity from the user

SR3.1.2.1.3.2. The VG system shall accept the torque threshold from the user

SR3.1.2.1.3.3. The VG system shall accept the default pixel offset from the user

SR3.1.2.1.3.4. The VG system shall accept the mode of assist from the user

SR3.1.2.1.3.5. The VG system shall accept the mode of adaptive performance from the user

SR3.1.2.1.4. The VG system shall enable user to navigate between the input menu and the game window

SR3.1.2.2. VG System output requirements

SR3.1.2.2.1. The VG system shall display static game elements on the game window

SR3.1.2.2.2. The VG system shall display dynamic game elements on the game window

SR3.1.2.2.2.1. The VG system shall display the current ball position on the game window

SR3.1.2.2.2.2. The VG system shall display the origin position of the ball on the game window

SR3.2. Internal interface requirements: N/A

SR4. System internal data requirements

SR4.1. The system shall store the input torque values as a vector

SR4.2. The system shall store the foot switch values as a vector

SR5. System Human Factors requirements

SR5.1. The VG system shall display the current torque value relative to the maximum torque value on the game window

SR5.2. The VG system shall display the points scored at the end of each gait cycle on the game window

SR5.3. The VG system shall display the cumulative score of the trial on the game window

SR5.4. The VG system shall clearly indicate when a goal is scored

SR6. System constraints

SR6.1. The ball movement in real time shall be given by

$$y_{ball} = \alpha \tau_{max} + y_{start}$$

SR6.1.1. The maximum torque shall be given by

$$\tau_{max,i+1} = \begin{cases} \tau_{max,i} , \tau_{i+1} \leq \tau_{max,i} \\ \tau_{i+1} , \tau_{i+1} > \tau_{max,i} \end{cases}$$

SR6.2. The patient performance shall be determined by the change in the start position of the ball -  $y_{start}^* = y_{start} + sgn[r]\Delta y_{start}$

SR6.2.1. The relative change in performance is given by

$$\Delta y_{start} = |\beta r|$$

SR6.3. The instantaneous score of the patient shall be determined by

$$p_i = \begin{cases} 0, & y_{ball} \leq b_1 \\ a_1, & b_1 \leq y_{ball} \leq b_2 \\ a_2, & b_2 \leq y_{ball} \leq b_3 \\ a_3, & b_3 \leq y_{ball} \leq b_4 \end{cases}$$

Table 5 defines the terms used in the constraint.

Data Dictionary	
$y_{ball}$	Position of the ball in positional coordinate system (unit – pixels)
$\alpha$	Torque-screen pixel sensitivity constant (unit – pixels/Nm)
$\tau_{max}$	Maximum human torque exerted during the trial (unit – Nm)
$\tau_i$	Instantaneous human torque exerted at the $i^{th}$ time instant (unit – Nm)
$y_{start}^*$	New start position of the ball at the beginning of a cycle (unit – pixels)
$y_{start}$	Start position of the ball at the beginning of a cycle (unit – pixels)
$r$	Co-relation coefficient between the peak human torque applied and time (unit – dimensionless)

$\beta$	Default pixel offset – a user determined pixel change in the screen for a 100% improvement in the human torque exerted during a trial (unit – dimensionless)
$\Delta y_{start}$	Change in the start position as indicated by an adaptive performance algorithm (unit – pixels)

Table 5: Data dictionary for the VG system

## 2.2. Candidate system architecture

Based on the system requirements generated, a candidate system of interest is developed which can be packaged as a single input (robot torque)—output (screen cursor) computer Video Game. It is based on the penalty kick scenario in a soccer game. The objective is to provide step-by-step bio-feedback during assisted-walking to facilitate patient's walking performance (e.g. swing DF clearance). A key feature of the game is that it challenges the patient based on their performance by incorporating an adaptive learning algorithm. A SysML Block Definition Diagram (BDD) is used to represent structural decomposition of the system of interest as shown in Fig 10. The video game system is a software system which has software, hardware and human interfaces. The three main elements of the system are

- User Input Menu: A graphical User interface that enables users to enter the game inputs
- Data acquisition and Game logic: The core game engine that contains the mathematical response model governing the values of metric(s) of interest

- Game progression logic: This consists of the mathematical algorithm responsible for the game difficulty level progression

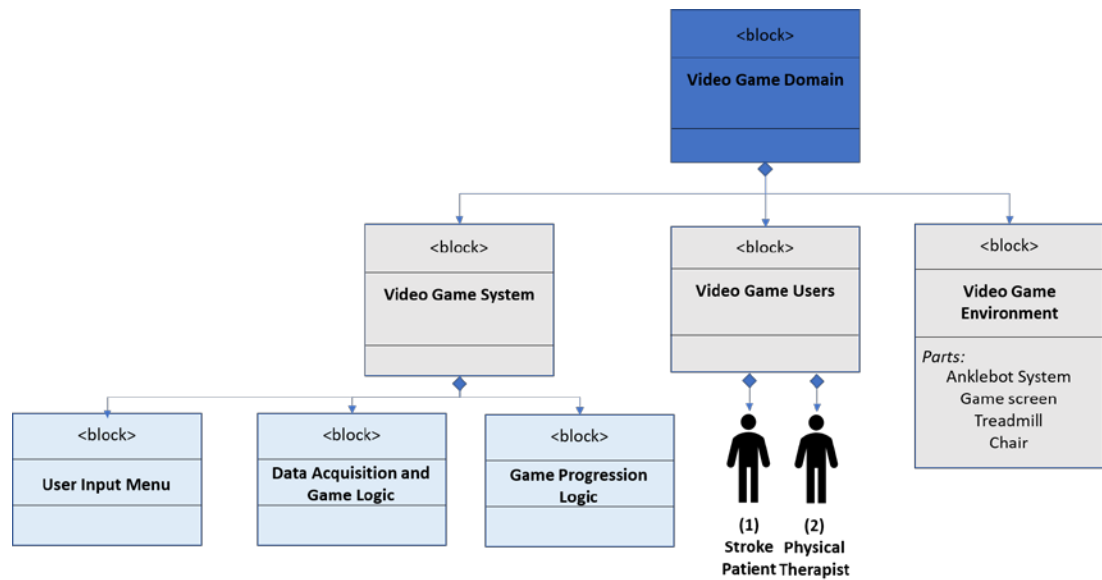


Figure 10. A Domain Block Definition Diagram (BDD) of the system describing the structural decomposition of the Video Game System

Simple information flow architecture of the game is shown in Fig. 11.

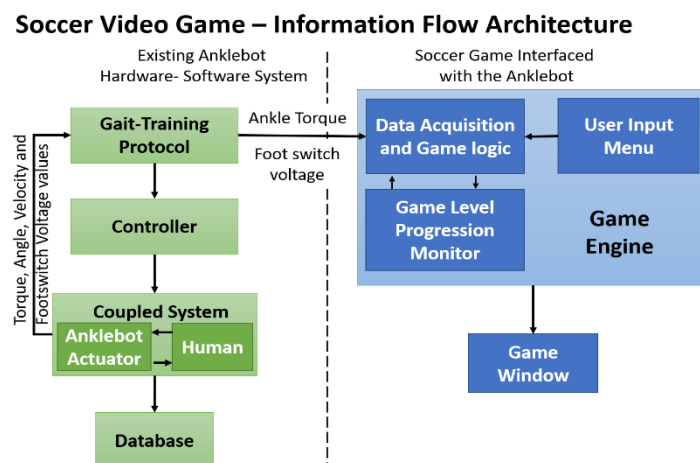


Figure 11. A high-level information flow architecture diagram of the game interfaced with the existing Anklebot hardware-software system.

A SysML internal block diagram (IBD) presented in Fig 12, shows the information flow between the Video game system with its environment.

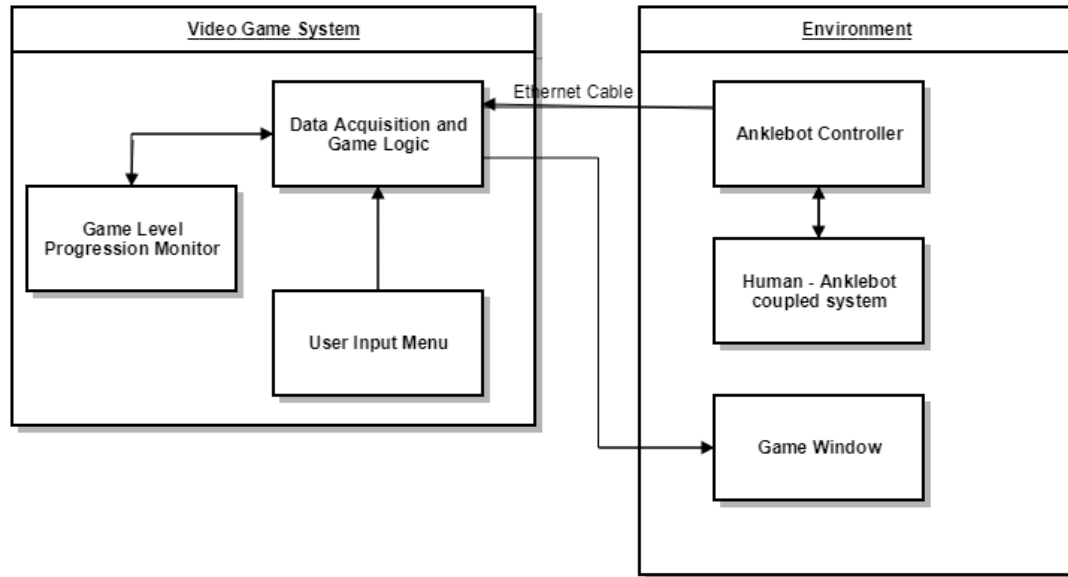


Figure 12. Internal Block Diagram describing the flow of information in the system

### 2.3. Mathematical response model of the system

A Response Model/Function is a mathematical or algorithmic model that predicts how a system performs as a function of its design characteristics and environmental characteristics. A response variable is the system performance measure of interest, and an explanatory variable is a variable that affects the system performance. When developing a model, one wants to determine how some output metric of interest ( $M$ ) depends on the values (controlled or uncontrolled) of a variety of input factors ( $F_i$ ). This can be used to guide the model development which translates into system realization (or synthesis) and

also in test design. Fig 13 shows the response model diagram for the VG system. Here, our metric of interest is the ball screen placement -  $y_{ball}$  and the factors affecting are – the instantaneous torque value  $\tau_{i+1}$ , the previous recorded maximum torque value of the cycle  $\tau_{max}$ , the torque-to-screen pixel sensitivity coefficient  $\alpha$ , and the foot switch value V to determine the trigger of the desired event.

The metric of interest or the Measure of Effectiveness of the System is the instantaneous ball screen position -  $y_{ball}$ .

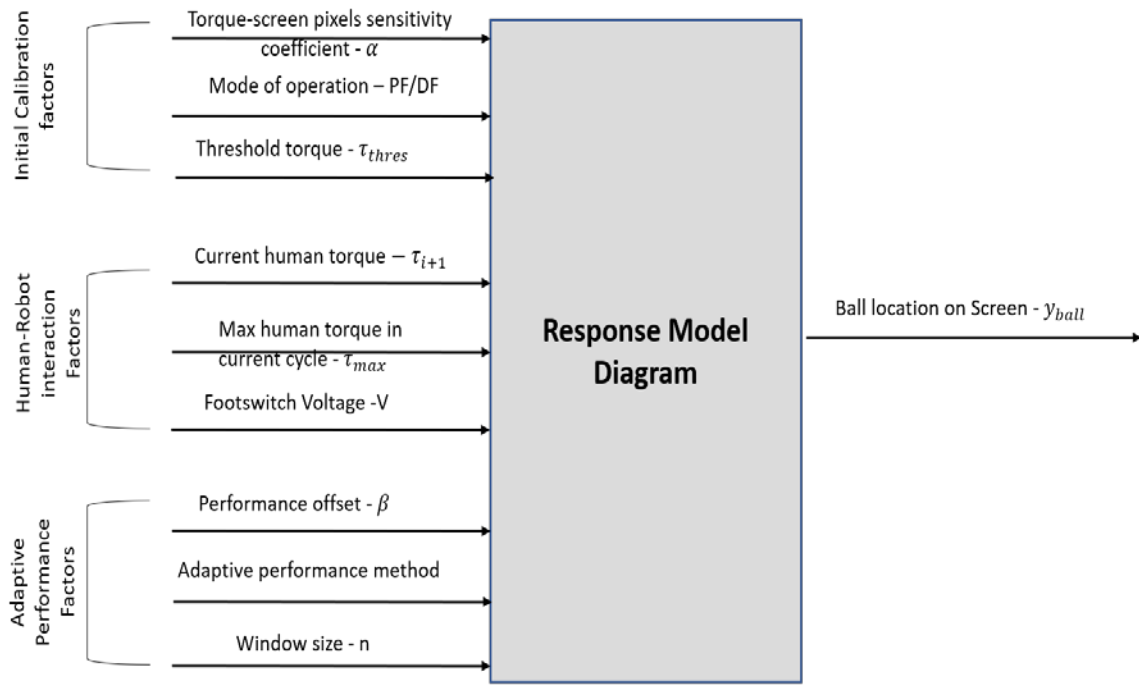


Figure 13. Mathematical response model of the VG system

The soccer game is event-triggered (i.e. activated only for desired events corresponding to gait deficits) with straight forward rules. The ball is initially placed on a red line and traverses linearly towards the goal as the patient starts the robot-assisted gait training exercise (Fig. 14). The game is designed to run in two modes:



- a) Dorsiflexion, to provide assist during the swing phase (characterized between the toe-off and foot-strike events). In this mode, the biomechanical metric of interest is the peak swing angle ( $\theta_{PSW}$ )
- b) Plantarflexion, to provide supplemental propulsive assist during the stance phase (characterized between the foot-strike and toe-off events). In this mode, the biomechanical metric of interest is the estimated peak human torque ( $\vec{\tau}_H$ ). To make intuitive sense, the PF mode consists of the origin of the ball in the middle of the screen with the ball moving downwards during the stride. Similarly, in the DF mode, the ball starts at the origin and travels upwards.



Figure 14. Soccer game rudiments window showing its UI.

The model development proceeds along the following lines: the net ankle torque ( $\vec{\tau}_{net}$ ) is the vector sum of the robot ( $\vec{\tau}_R$ ) and human ( $\vec{\tau}_H$ ) vector torques i.e.

$$\vec{\tau}_{net} = \vec{\tau}_H + \vec{\tau}_R. \quad (1)$$

Since the ankle robot is controlled by a simple impedance controller, the net robot torque may be expressed as:

$$\vec{\tau}_{net} = K\theta + B\dot{\theta}, \quad (2)$$

where K and B are the programmable controller stiffness and programmable controller damping, respectively. Compensating for device stiction ( $\pm 1.41$  Nm), we can estimate the human torque ( $\widehat{\tau}_H$ ) in absence of a force transducer as:

$$\widehat{\tau}_H = K\theta + B\dot{\theta} - \vec{\tau}_R - \vec{\tau}_{stiction}, \quad (3)$$

The game screen is symmetric about the vertical axis. The initial geometric position of the game screen elements described by their vertical position in pixels is expressed as:

$$y_{start} = \frac{y_{max} - y_{min}}{2} \quad (4)$$

where,  $y_{start}$  is the starting position of the red line indicating the origin position of the ball at the start of each trial of the game,  $y_{max}$  is the position of the goalpost,  $y_{min}$  is the farthest position that the line can take without crossing the bounds of the screen. The movement range of the ball,  $H$  is defined as:

$$y_{max} - y_{min} = H \quad (5)$$

In-sole foot switches that generate a discrete cumulative foot-switch voltage determine the “state” of the videogame (i.e. active functionality, or not). The DF or PF human ankle torque causes movement of the screen cursor. Hence,

$$y_{ball} = \alpha \widehat{\tau}_H + y_{start} \quad (6)$$

where,  $y_{ball}$  is the absolute position of the ball during the gameplay and  $\alpha$  is a sensitivity scaling constant (pixels/Nm) that relates  $\tau_H$  to  $y_{ball}$ . In practice, we conduct a seated

isometric test before the actual TM based gait training to evaluate the patient's maximum torque capability (refer to section Use Case Narrative 1 in section 2.4), thus enabling the therapist to set the sensitivity  $\alpha_{calib}$  for each patient. This performed by patient exerting maximum volitional DF or PF torque ( $\tau_{H|max}$ ) against a “stiff” spring whose impedance is characterized by the controller programmable stiffness and programmable damping. Hence,

$$\alpha_{calib} = \frac{H/2}{\tau_{H|max}} \quad (7)$$

The underlying algorithm kicks in when the footswitches indicate an event of interest. The algorithm checks for the peak estimated human torque during phase of interest (e.g. stance for PF assist or swing for DF assist) and continually updates per the following rule:

$$\tau_{max,i+1} = \begin{cases} \tau_{max,i} , \tau_{i+1} \leq \tau_{max,i} \\ \tau_{i+1} , \tau_{i+1} > \tau_{max,i} \end{cases} \quad (8)$$

with  $\tau_{max,1} = \tau_1$ . Hence, for each sample during the phase of interest, the position of the ball is given by:

$$y_{ball} = \alpha \tau_{max} + y_{start} \quad (9)$$

In the seated training protocol of the Anklebot using the “racer game” [28,29], the game parameters did not dynamically adapt to human performance within an ongoing trial. In the soccer game for robot-assisted gait training, we have embedded an adaptive performance algorithm that dynamically auto-adjusts videogame parameters (e.g. difficulty indexed by  $y_{start}$ ) based on ongoing improvement or decline in the patient's performance within the same robot-assisted session. We embed two methods—a simple average or the correlation coefficient ( $r$ ) of the peak human torque ( $\widehat{\tau_H}$ ) across individual gait cycles. To be specific, the difficulty is increased (or decreased) by moving the cursor (ball) starting line further

away (or closer) from the goal post. The extent to which the starting line is moved depends on the relative improvement (or decline) in performance over a user-specified fixed analysis window ( $N$ ), which is numerically calculated either using an average over past ( $N-1$ ) cycles compared to the  $N^{\text{th}}$ -cycle, or by examining the sign and magnitude of the correlation coefficient ( $r$ ). For instance, for the regression method, the start line is moved away or towards from the goal post depending on improvement ( $r > 0$ ) or decline ( $r < 0$ ), respectively as shown in Fig. 15. The new position of the starting line ( $y_{start}^*$ ) is given by:

$$y_{start}^* = y_{start} + \text{sgn}[r]\Delta y_{start} \quad (10)$$

$$\Delta y_{start} = |\beta r|.$$

where  $\beta$  is user-specified scaling factor (performance offset) that amplifies the improvement (or decline) to a change in the starting line

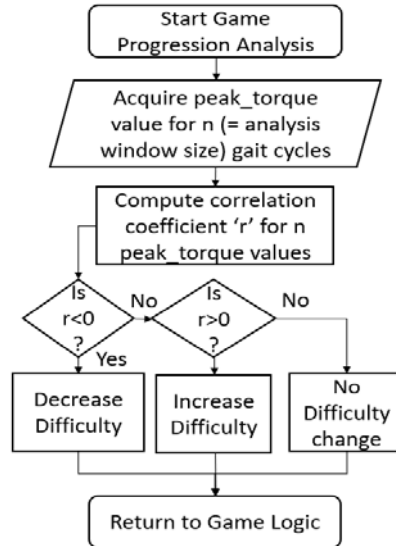


Figure 15. Flowchart describing the adaptive performance algorithm using the correlation coefficient as the determinant parameter. Difficulty refers to starting line position with respect to the goal post, as defined by Eq. (10). A higher value indicates further away from the goal post (i.e. more difficult or challenging), and vice versa.

## Chapter 3. Game Development

### 3.1. Lifecycle model of the Software Development of the Video Game

The Life Cycle diagram for the whole project is chosen to be the V Lifecycle Model. The V model was chosen as it has a sequential process flow like the Waterfall model. The V model is a test-driven model where the testing strategy is even before the code. This way, the defects or bugs are identified very early in the life cycle. The V Lifecycle model for this project can be found in the Fig 16.

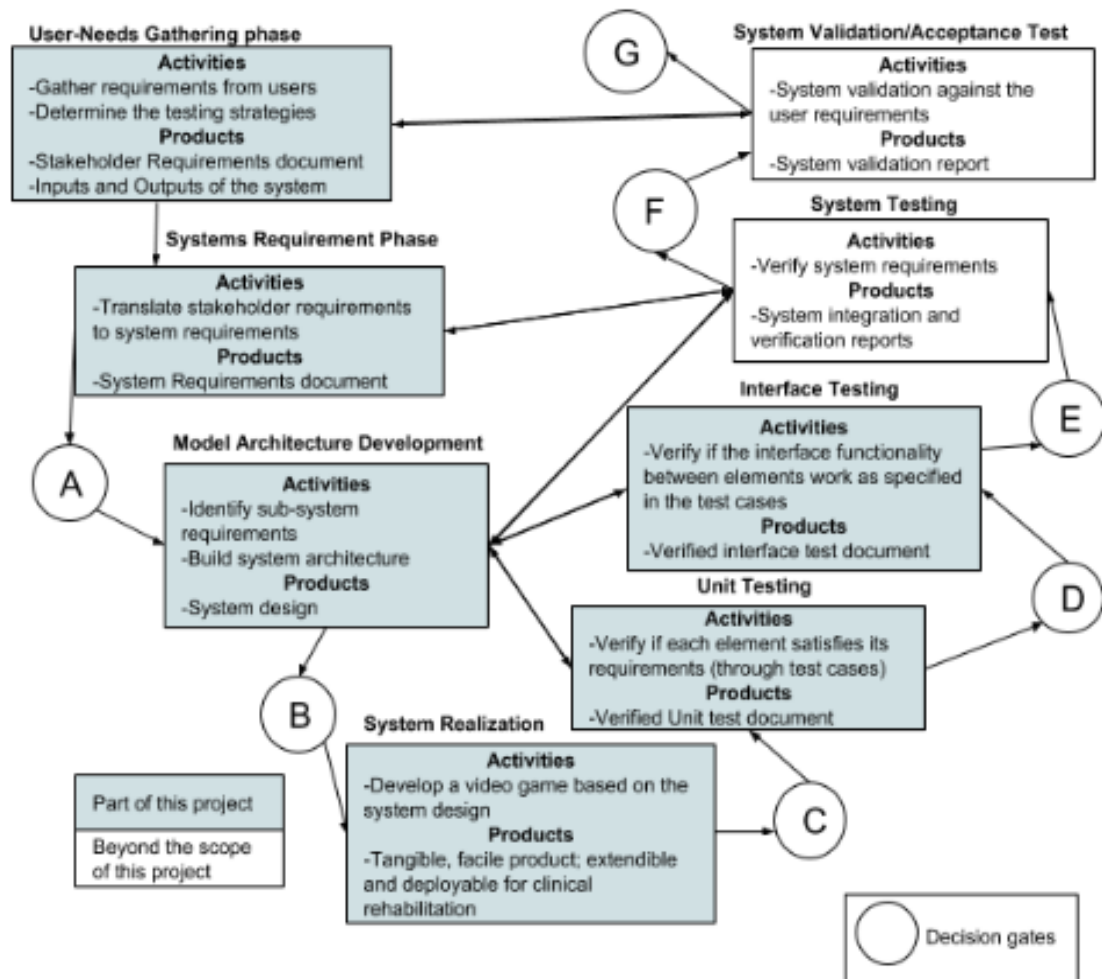


Figure 16. V Development Lifecycle model of the VG System

The user needs are gathered from the extensive background study and literature review described in Chapter 1. The stakeholder requirements and model development are discussed in Chapter 2. Unit and Interface testing are discussed in Chapter 4. System Realization phase is discussed in this chapter.

### 3.2. System realization

The System Realization phase in itself can be considered as a sub-project which requires multiple iterations. The team, decided to use a spiral model for this phase. The spiral model is very useful in creating a new design. It allows incremental modelling that adds on functionality, fidelity and accuracy in subsequent iterations. The first iteration would consist of a very simplified Model. Table 6 summarizes the various iterations in this model. Each iteration added an incremental functionality to the Software System.

<b>Iteration Number</b>	<b>Incremental Functionality</b>
1	Keyboard input moves the ball up and down
2	A dummy vector of “offline” positions moves the ball up and down
3	The mathematical model established in Chapter 2 responsible for on screen movement of the ball. Driven by real offline data
4	Adaptive performance algorithm embedded in the software system
5	Interfacing with the user input menu for initial settings
6	Interfacing with the physical Anklebot System for real time “online” functionality

Table 6: Summary of each iteration in the Spiral of the System Realization phase

The Life Cycle model shown in Figure 18, has multiple decision gates. Decision gates are the points in the lifecycle where the development team show their work to the stakeholders

(in this case an agreed meeting of the complete research team) to see if the project path is agreeable to both parties. The following is the description of the decision gates.

- Decision Gate A: This is the step where the development team show the stakeholders what they think to be the System requirements. Project shall proceed further only when the stakeholders and developers agree upon a set of system requirements.
- Decision Gate B: This is the step where the developers show the stakeholders the detailed system architecture. Project shall proceed further only when the stakeholders and designers agree upon a System Architecture.
- Decision Gate C: This is the step where the developers show the stakeholders the Realized VG System. Project shall proceed further only when the stakeholders agree upon the this Realized System.
- Decision Gate D: This is the step where the developers test each of the elements of the system. The developer shall not proceed till all the elements pass their unit tests. If a unit test is failed, the developer changes the Model implementation appropriately till the test is passed.
- Decision Gate E: This is the step where the developers test if VG system can be successfully interfaced (Hardware and Software) with the Hardware and Software of the Anklebot System. The developers shall not proceed till the VG system successfully interfaces with the Anklebot System. If the interface test is failed, the developer changes the Model implementation and the interface strategy appropriately till the test is passed.

- Decision Gate F: This is the step where the developers perform a complete system test. The developers shall not proceed till all the System test cases as generated by the System Requirements are passed. If the system test is failed, the developer can change the System architecture or the element architecture or the system realization strategy appropriately till the test is passed. This gate is beyond the scope of this research.
- Decision Gate G: This is the step where the developers and stakeholders validate the system against the user needs to check whether the product created satisfies the stakeholder requirements.

The products at the end of each of the stages of the V- lifecycle are explained as below:

- Customers' needs gathering phase – This is the stage in the V-Lifecycle where the emphasis is on understanding, aggregating and formally representing the needs of the customer. The products developed at the end of this stage are:
  - Stakeholder requirements table: This refers to a tabulated representation of the desired requirements of the stakeholder as agreed by both the customer and the Systems Analyst. The structure of every requirement is simple and unambiguous.
  - List of inputs and outputs to the system: This refers to the formal understanding of what goes into the system – inputs (or factors) and what comes out of the system – outputs (or metrics). This output consists of a list of the inputs to the system and the outputs to the system along with their associated units.



- Systems Requirement phase - This is the stage in the V-Lifecycle where the emphasis is on translating the customer needs into the system needs and requirements. The products developed at the end of this stage are:
  - Systems Requirements table: This refers to a tabulated representation of the desired requirements that the system should satisfy, as agreed by both the customer/stakeholder and the Systems Analyst. The structure of every requirement would be simple and unambiguous. The requirements can be further classified as those derived from the main requirements.
- Model Architecture Development phase – This is the stage in the V-Lifecycle where the emphasis is on developing the architecture and thus a guide map to the actual model creation. The products developed at the end of this stage are:
  - System Design: This refers to the SysML diagrams that describe the architecture – both structural and functional, of the system of interest, which in this case is the VG System.
- System Realization Phase – This is the stage in the V-Lifecycle where the emphasis is on actual development of the Software System. In our case, of the VG system, Python is the platform used to realize the system. The System is realized (programmed) strictly in adherence to the outputs of the Model Architecture Development phase. Sub modules of Python – pygame and tkinter are used primarily for the development.
  - Pygame: Pygame is a cross-platform set of Python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used with the Python programming language. It is built over

a simple DirectMedia Layer (SDL) library, with the intention of allowing real-time computer game development without low-level mechanics of C programming language and its derivatives. This makes it ideal to be used for the graphical display of the Video Game developed.

- Tkinter: Tkinter is a Python binding to the Tk GUI toolkit. It is a standard Python interface to the Tk GUI toolkit and is Python's de facto standard GUI. For the purposes of simplicity, the user input was designed using Tkinter.
- Unit Testing phase – This stage of the V-Lifecycle refers to the stage where each of the elements are tested according to the test plan laid out by the unit test cases/strategies. The product at this stage is:
  - Verified unit test document: This refers to a document that compares the expected output values from an element for certain inputs. If the actual output is comparable to the expected output within a pre-set region of acceptable tolerance, that particular test case is said to be passed. A verified unit test document is the one that contains an aggregation of the results of all the unit tests.
- Interface Testing phase – This stage of the V- Lifecycle refers to the stage where the interface characteristics between the VG system and the Anklebot system (Hardware and Software Sub systems) is tested according to the test plan laid out by the interface test cases. The product at this stage is:
  - Verified interface test document: This refers to a document that compares the expected output values from interfacing between the VG system and the

Anklebot System for certain inputs. If the actual output is comparable to the expected output within a pre-set region of acceptable tolerance, that particular test case is said to be passed. A verified interface test document is the one that contains an aggregation of the results of all the interface tests.

System testing and validation testing require additional Human Subject Research approval and a prolonged testing duration for clinical considerations. Hence, it is out of the scope of this research.

## **Chapter 4. Interfacing and Verification**

### **4.1. Hardware interfacing**

The Anklebot system and the VG system were interfaced through a Local Area Network (LAN). A temporary LAN connection was established through a physical connection between the Anklebot computer and the Personal computer running the VG software system. Both the computers were running a Linux based operating system. RJ45 connector was used for this interfacing. A registered jack (RJ) is a standardized physical network interface for connecting telecommunications or data equipment. The physical connectors that registered jacks use are mainly of the modular connector and 50-pin miniature ribbon connector types. The most common twisted-pair connector is an 8-position, 8-contact (8P8C) modular plug and jack commonly referred to as an RJ45 connector.

### **4.2. Software interfacing**

The software interfacing was an interesting challenge to tackle. The Anklebot System consists of a self-sufficient, robust software system. With one of the primary requirements being development of the VG System as a standalone software, utmost care had to be taken care to make the interfacing as minimal as possible without compromising on the function and reliability. A level of abstraction about the Anklebot software system had to be maintained while developing the VG system. This was necessary to make minimal modifications to the Anklebot software system to aid in the interfacing.

After experimenting with multiple strategies, a twofold algorithm was decided to enable the software interfacing for information flow.

- The Gait training program of the Anklebot software system was slightly modified to display the instantaneous torque and the footswitch voltage values on a terminal window on the screen.
- Through the use of an open source networking tool called *Netcat*, the values displayed on the terminal window of the Anklebot computer screen were pipelined into the terminal window of the PC running VG system. A simple Python script was written to capture this continuous stream of information and store it into an array. This array was fed into the game in real time.

Netcat is a versatile tool often regarded as the ‘swiss knife of networking’. Netcat can be used to relay data in real time over a network through multiple machines too. Data can be encrypted using various hashing methods over this network. This is shown in Fig 17. For the purpose of prototyping, a single machine interface is currently used with no data encryption. The existing framework can easily be augmented to accommodate multiple machines transmitting encrypted data. This could potentially be useful if the stakeholders decide to introduce a multiplayer game functionality training multiple similarly disabled patients at a time.

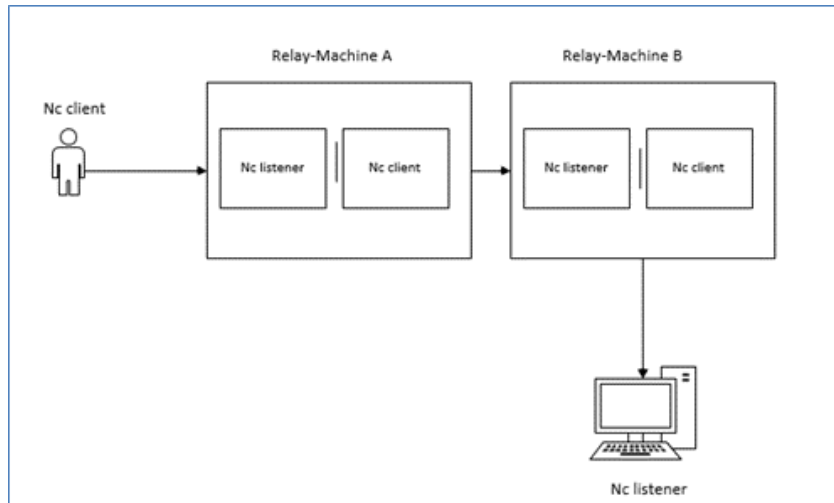


Figure 17. A conceptual flow diagram explaining the information flow using the Netcat(Nc) tool. Nc client is the computer that is transmitting the data which in our case is the Anklebot computer. Nc listener is the PC running the VG System.

Netcat uses the internet protocol suite – commonly known as the TCP/ IP communication protocol. The Internet protocol suite is the conceptual model and set of communications protocols used on the Internet and similar computer networks. It is commonly known as TCP/IP because the foundational protocols in the suite are the Transmission Control Protocol (TCP) and the Internet Protocol (IP). It is occasionally known as the Department of Defense (DoD) model, because the development of the networking method was funded by the United States Department of Defense through DARPA.

The Internet protocol suite provides end-to-end data communication specifying how data should be packetized, addressed, transmitted, routed, and received. This functionality is organized into four abstraction layers which classify all related protocols according to the scope of networking involved. From lowest to highest, the layers are the link layer, containing communication methods for data that remains within a single network segment

(link); the internet layer, providing internetworking between independent networks; the transport layer handling host-to-host communication; and the application layer, which provides process-to-process data exchange for applications.

#### **4.3. Verification**

##### **4.3.1. Unit Testing**

In computer programming, unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. Intuitively, one can view a unit as the smallest testable part of an application. In procedural programming, a unit could be an entire module, but it is more commonly an individual function or procedure. The VG system is a relatively simplistic system, in which unit testing was performed by line by line review of the code and also using dummy inputs to check if the outputs matched the predicted outputs that given by the mathematical response model.

##### **4.3.2. Interface Testing - HRI Video Game Validation – “Causality”**

Stiction-compensated peak torque values were used as an offline input to stimulate the videogame. The real-time on-screen ball position was predicted by Eqs. (8) and (9) within and across gait cycles and was in concordance with desired behavior (Fig. 18)—either the ball position saturated with respect to the previous position (when differential torque did not increase) or moved closer to the goal post (when differential torque increased).

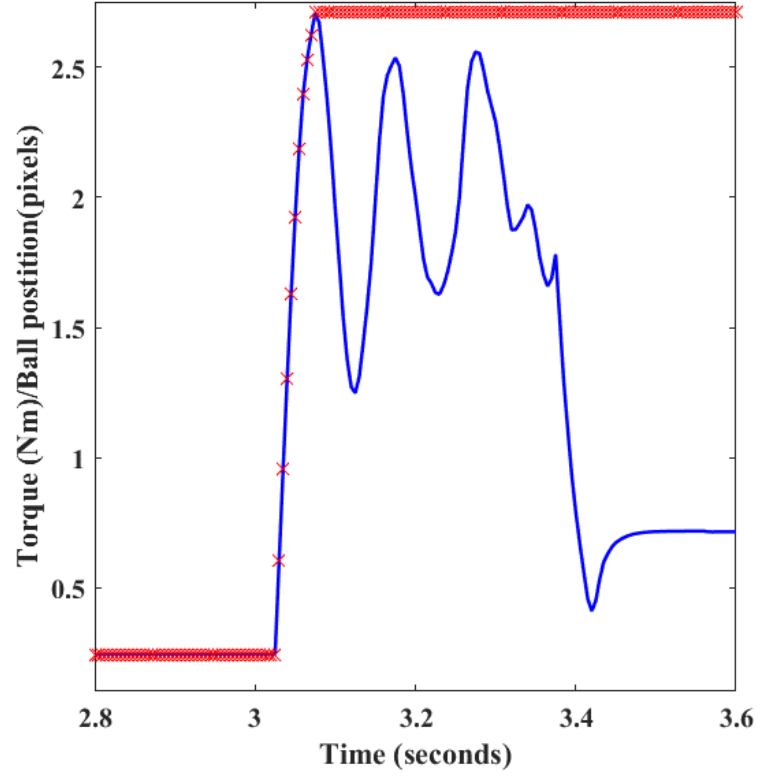


Figure 18. Input – Output characterization of the HRI – video game system. The ball position conforms to Eqs (8) and (9) for  $\alpha = 0$

#### 4.3.3. Physical signal validation

Data was collected from the robot placing it on the bench (i.e. no human) programmed for DF deficit (target event: toe-off) and replicating gait cycles by repeatedly stepping on and off the footswitches. This was done to verify that the torque profile corresponds to the commanded angle trajectory. Figures 19 shows approximate angular acceleration profiles from ankle angle data. We then compared the net torque profile ( $\vec{\tau}_{net} \approx \vec{\tau}_R, \vec{\tau}_H = 0$ ) from Eq. (3) to verify that the sign of the stiction-compensated torque and angular acceleration are in close resemblance. The stiction-compensated torque was filtered using a Lowess smoothing filter in Matlab.



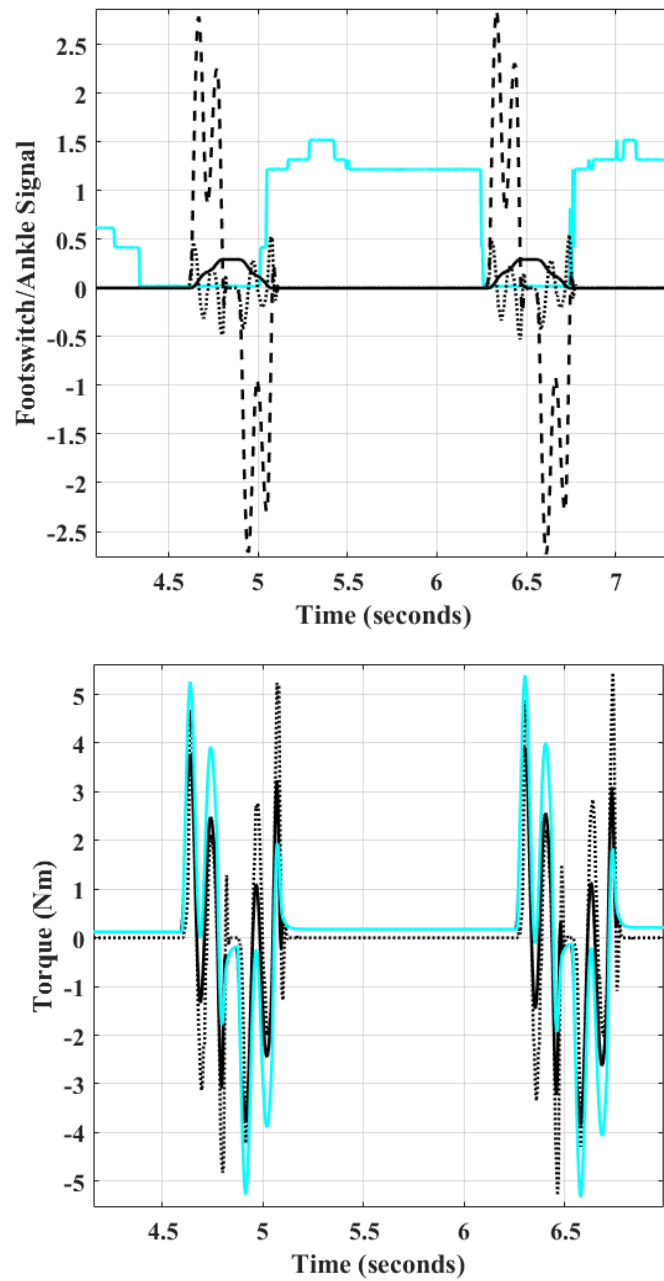


Figure 19. Bench tests of the human-robot interface with event-triggered robotic actuation (Top) Raw traces of footswitch voltage (—), ankle angle (—), velocity (—), and angular acceleration  $\times 0.005$  (...); (Bottom) Comparison of stiction-compensated torques vs. scaled angular acceleration  $\times 10$ , both shown for two simulated gait cycles.

#### 4.3.4. Adaptive performance Validation

A simple experiment was designed to test the adaptive performance algorithm. The parameter of choice for this experiment was the correlation coefficient. A sample gait trial was recorded with varying performance. This data was fed into the VG System. Correlations were performed every 5 cycles—if the coefficient was positive (i.e. improvement, or  $sgn[r] > 0$ ), the start line position increases ( $\Delta y_{start}$ ) (i.e. moves away from the goal post providing a challenge). Similar, if the coefficient was negative (i.e. decline, or  $sgn[r] < 0$ ), the start line position decreases ( $\Delta y_{start}$ ) (i.e. moves closer to the goal post providing less difficulty). Figure 21 shows an example when there was consistent decline in performance in human torque ( $\widehat{\tau}_H$ ), causing adjustments to the task difficulty (i.e. monotonic decrease in the starting line with respect to the goal position). Fig 20 shows a graph of the start line distance as a function of time.

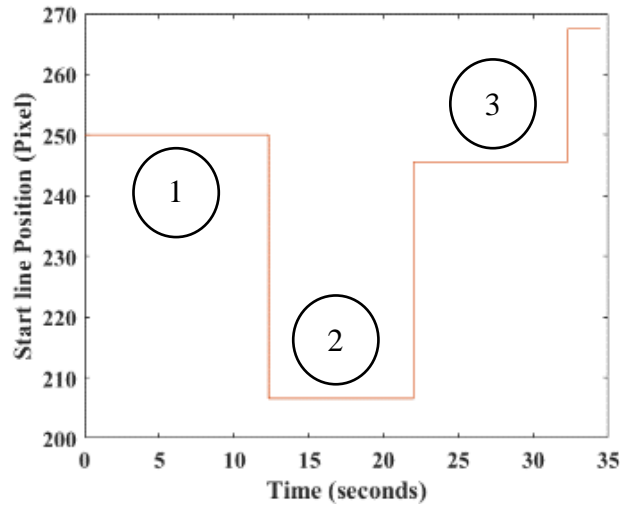


Figure 20. Start line distance v/s Time using an Adaptive performance algorithm

Adaptive performance algorithm mediated (correlation method) adjustments to the task difficulty (starting line) with changes in peak human torque across gait cycles. Starting line is initially set at 250 pixels and correlations are computed over an analysis window  $N=5$  cycles (shown by region “1”) and  $N=8$  cycles (shown by region “2”). This causality between the change in start line distance and a decrease in performance indexed by a negative correlation coefficient  $-r$ , can be clearly seen in Fig 23.

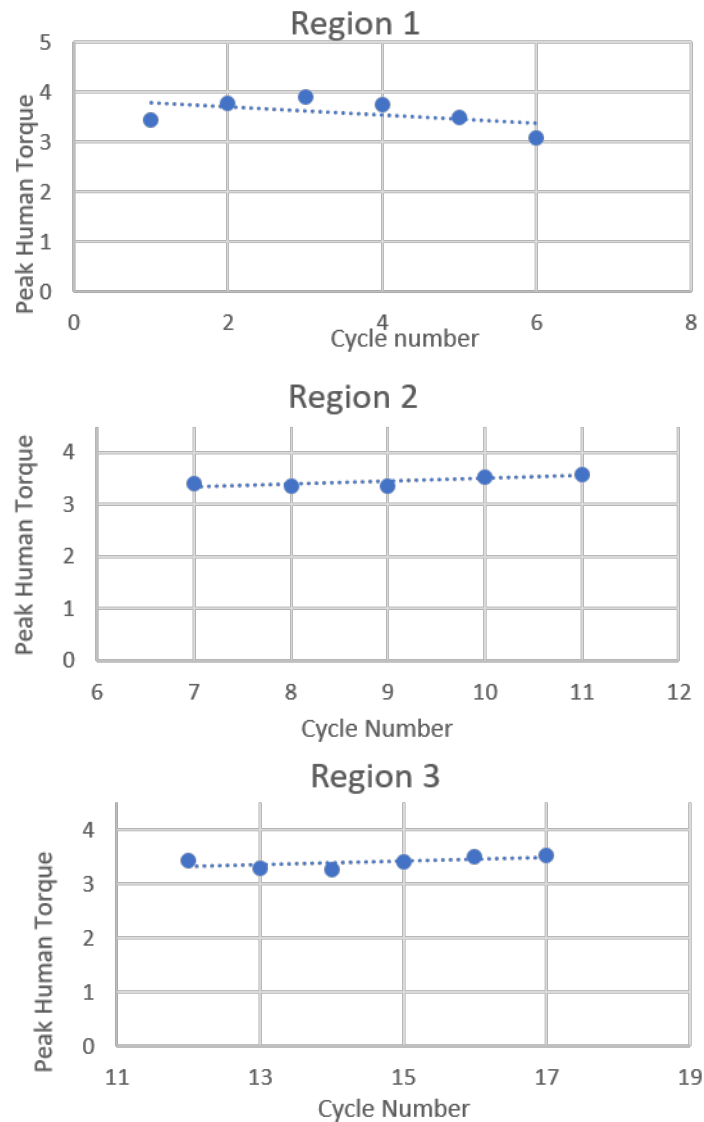


Figure 21. Start line distance v/s Time using an Adaptive performance algorithm

## **Chapter 5. Conclusions and Future work**

The work presented here can be summarized to be a stand-alone, functional, extendable, software platform based on well established and published research in the fields of biomechanics and rehabilitation robotics. Model Based Systems Engineering approach has been adopted to facilitate an easy, robust and traceable development and integration, and deployment and maintenance in the future. This software has been successfully tested in a few of the key functions. It is now ready for a formalized system usability testing to ensure robustness and adherence to the derived requirements.

Future work in this direction would involve a formal system validation or a usability test in its operating environment – the Anklebot system. All the tests conducted so far have been “bench-testing” methods, i.e. without a human in the loop. The usability test would consist of a small set of healthy individuals validating efficacious use of the game as a motivational bio stimulus guiding the Anklebot assisted gait training therapy protocol. This would answer the questions of whether the game is synchronous with the walking speed of an individual and also tolerate step-step variability in timing, and also whether the choice of game selection – a soccer scenario provides an acceptable stimulus.

Post successful completion of a usability test, the developed system can be integrated in clinical trials evaluating the benefits of robot assisted gait training for geriatric stroke survivors. The results of this clinical testing can be used to test the overarching hypothesis, foundational to this thesis – “Interactive visual task, when integrated with Treadmill Training, will further increase the ankle neuro-motor and whole-body functional gains (and learning rates in those measures), and these performance changes can shape robotic outputs to close the loop”.

Potential improvements to the game model would be to incorporate machine learning algorithms for real-time performance adaptation of the game to increase or decrease the difficulty. The research team is currently discussing about the possibilities of extending this framework for a strength training setting. The underlying mathematical response model and the software development is robust enough to extend this functionality of for other rehabilitative ankle robots, any joint specific rehabilitative exercise device and also for exercise devices targeted at strength training of muscles.

An engineering development would be to improve the user experience of the game by incorporating better visual aids and following design principles of game development. Packaging this game into an executable mobile or desktop application would be a good step to ensure the technologically agnostic end users – physical therapists and end users remain abstracted from the white-box view of the system.

This computer-based video game could be extended to inversion-eversion training in the frontal plane or even a combinatory movement of dorsi-plantarflexion and inversion-eversion, to facilitate whole body gains.

I envision this research to lead to a suite of rehabilitative products that could be used as an add-on to any existing joint training protocol with a variety of motivational stimulus through visual bio-feedback.

## References

- [1] Heart Disease and Stroke Statistics – 2018 update, American Heart Association.  
<http://www.americanheart.org/statistics>.
- [2] Effects of stroke: American Heart Association  
[http://www.strokeassociation.org/STROKEORG/AboutStroke/EffectsofStroke/Effects-of-Stroke\\_UCM\\_308534\\_SubHomePage.jsp](http://www.strokeassociation.org/STROKEORG/AboutStroke/EffectsofStroke/Effects-of-Stroke_UCM_308534_SubHomePage.jsp).
- [3] S. J. Olney and C. Richards, “Hemiparetic gait following stroke. Part I: Characteristics,” *Gait Posture*, vol. 4, pp. 136-48, 1996.
- [4] M. G. Bowden, C. K. Balasubramanian, R. R. Neptune, and S. A. Kautz, “Anterior-posterior ground reaction forces as a measure of paretic leg contribution in hemiparetic walking,” *Stroke*, vol. 37, pp. 872-76, 2006.
- [5] S. J. Olney, M. P. Griffin, T. N. Monga, and I. D. McBride, “Work and power in gait of stroke patients,” *Arch Phys Med Rehabil.*, vol. 72, pp. 309-14, 1991.
- [6] A. Ramnemark, L. Nyberg, B. Borssén, T. Olsson, and Y. Gustafson, “Fractures after stroke,” *Osteoporos Int*, vol.8, pp. 92-95, 1998.
- [7] M. S. Dennis, K. M. Lo, M. McDowall, and T. West, “Fractures after stroke: frequency, types, and associations,” *Stroke*, vol. 33, pp. 728-34, 2002.
- [8] M. Hallett, “Plasticity of the human motor cortex and recovery from stroke,” *Brain research reviews*, vol. 36(2), pp. 169-74, 2001.
- [9] J. D. Schaechter, *Motor rehabilitation and brain plasticity after hemiparetic stroke*, *Progress in neurobiology*, vol. 73(1), pp. 61-72, 2004.
- [10] J. W. Krakauer, “Motor learning: its relevance to stroke recovery and neurorehabilitation,” *Current opinion in neurology*, vol. 19(1), pp. 84-90, 2006.

- [11] L. W. Forrester, A. Roy, C. Hafer-Macko, H. I. Krebs, and R. F. Macko, "Task-Specific Ankle Robotics Gait Training After Stroke: A Randomized Pilot Study," *J NeuroEng Rehabil.*, vol. 13, pp. 51, 2016
- [12] R. F. Macko, F. Ivey, L. W. Forrester, D. F. Hanley, J. D. Sorkin, L. I. Katzel, K. H. Silver, and A. P. Goldberg, "Treadmill Training Improves Fitness Ambulatory Function and Cardiovascular Fitness in Chronic Stroke Patients," *Stroke*, vol. 36, pp. 2206-11, 2005
- [13] R. F. Macko, F. Ivey, and L. W. Forrester, "Task-Oriented Aerobic Exercise in Chronic Hemiparetic Stroke: Training Protocols and Treatment Effects," *Top Stroke Rehabil.*, vol. 12, pp. 45-47.35, 2005.
- [13] M. Pohl, C. Werner, M. Holzgraefe, G. Kroczeck, J. Mehrholz, I. Wingendorf, G. H'olig, R. Koch, and S. Hesse, "Repetitive locomotor training and physiotherapy improve walking and basic activities of daily living after stroke: A single-blind, randomized 73ulticenter trial (Deutsche GangtrainerStudie: DEGAS)," *Clin. Rehabil.*, vol. 21, pp. 17-27, Jan.2007.
- [14] J. Mehrholz, C. Werner, J. Kugler, and M. Pohl, "Electromechanical assisted training for walking after stroke," *Cochrane Database Systematic Reviews*, vol. 4, pp. 5-8, 2007
- [15] L. Lunenburger, G. Colombo, R. Riener, R., and V. Dietz, "Biofeedback in gait training with the robotic orthosis Lokomat," *Proc of IEEE Int. EMBS Conf.*, vol. 2, pp. 4888-91, 2004.

- [16] S. K. Banala, S. H. Kim, S. K. Agrawal, and J. P. Scholz, "Robot assisted gait training with active leg exoskeleton (ALEX)," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 17(1), pp. 2-8, 2009.
- [17] H. Schmidt, S. Hesse, R. Bernhardt, and J. Krüger, "HapticWalker -a novel haptic foot device," *ACM Transactions on Applied Perception*, vol. 2(2), pp. 166-80, 2005.
- [18] Ekkelenkamp, R., Veneman, J., & van der Kooij, H "LOPES: a lower extremity powered exoskeleton," *Robotics and Automation, IEEE International Conference*, pp. 3132-3133, 2007
- [19] Bosecker, C. J., & Krebs, H. I, "MIT-skywalker," *Rehabilitation Robotics, ICORR 2009* pp. 542-549, 2007
- [20] Krebs, H. I., Hogan, N., Aisen, M. L., & Volpe, B. T, "Robot-aided neurorehabilitation," *IEEE transactions on rehabilitation engineering*, vol 6(1), pp.75-87.
- [21] Hogan, N., Krebs, H. I., Charnnarong, J., Srikrishna, P., & Sharon, A "MIT-MANUS: a workstation for manual therapy and training," *Proceedings., IEEE International Workshop on Robot and Human Communication*, pp. 161-165, 1992
- [22] Hogan N. Impedance Control: An Approach to Manipulation: Part I—Theory. ASME. *J. Dyn. Sys., Meas., Control*, vol. 107(1), pp. 1-7, 1985
- [23] Hogan N. Impedance Control: An Approach to Manipulation: Part II—Implementation. ASME. *J. Dyn. Sys., Meas., Control*, vol. 107(1), pp. 8-16, 1985
- [24] Hogan N. Impedance Control: An Approach to Manipulation: Part III—Applications. ASME. *J. Dyn. Sys., Meas., Control*, vol. 107(1), pp. 17-24, 1985



- [25] E. L. Miller, L. Murray, L. Richards, R. D. Zorowitz, T. Bakas, P. Clark, and S. A. Billinger, "American Heart Association Council on Cardiovascular Nursing and the Stroke Council. Comprehensive overview of nursing and interdisciplinary rehabilitation care of the stroke patient: a scientific statement from the American heart Association," *Stroke*, vol. 41, pp. 2402-48, 2010
- [26] VA/DOD Clinical practice guideline for the management of stroke. *J Rehabil Res Dev.*, vol. 47, pp. 1-43, 2010.
- [27] A. Roy, H. I. Krebs, D. J. Williams, C. T. Bever, L. W. Forrester, R. F. Macko, and N. Hogan. "Robot-aided neurorehabilitation: a robot for ankle rehabilitation," *IEEE Trans Robotics*, vol. 25, pp. 569-82, 2009.
- [28] L. W. Forrester, A. Roy, H. I. Krebs, and R. F. Macko, "Ankle Training With a Robotic Device Improves Hemiparetic Gait After a Stroke," *Neurorehabil Neural Rep.*, vol. 25, pp. 369-77, 2011.
- [29] L. W. Forrester, A. Roy, A. Krywonis, G. Kehs, H. I. Krebs, and R. F. Macko, "Modular Ankle Robotics Training in Early Sub-Acute Stroke: A Randomized Controlled Pilot Study," *Neurorehabil Neural Rep.*, vol. 28, pp. 678-87, 2014.
- [30] L. W. Forrester, A. Roy, C. Hafer-Macko, H. I. Krebs, and R. F. Macko, "Task-Specific Ankle Robotics Gait Training After Stroke: A Randomized Pilot Study," *J NeuroEng Rehabil.*, vol. 13, pp. 51, 2016
- [31] Hsu, A. L., Tang, P. F., and Jan, M. H., "Analysis of impairments influencing gait velocity and asymmetry of hemiplegic patients after mild to moderate stroke," *Archives of physical medicine and rehabilitation*, vol. 84(8), pp.1185-1193, 2003

- [32] Roy, A., Krebs, H. I., Williams, D. J., Bever, C. T., Forrester, L. W., Macko, R. M., and Hogan, N., “Robot-aided neurorehabilitation: a novel robot for ankle rehabilitation,” *IEEE transactions on robotics*, vol. 25(3), pp. 569-582, 2009
- [33] Brockett, Claire L., and Graham J. Chapman. “Biomechanics of the Ankle.” *Orthopaedics and Trauma*, vol. 30.3, pp. 232–238, 2016.
- [34] Grimston S.K., Nigg B.M., Hanley D.A., Engsberg J.R.” Differences in ankle joint complex range of motion as a function of age,” *Foot Ankle Int.*, vol. 14, pp. 215–222, 1993
- [35] Stauffer R.N., Chao E.Y., Brewster R.C.” Force and motion analysis of the normal, diseased, and prosthetic ankle joint,” *Clin Orthop Relat Res*, vol. 127, pp. 189–196, 1997.
- [36] Nordin M., Frankel V.H. Lippincott Williams & Wilkins, *Basic biomechanics of the musculoskeletal system*, 2001
- [37] A. Roy, H. I. Krebs, J. E. Barton, R. F. Macko, and L. W. Forrester, “Anklebot-assisted locomotor training after stroke: A novel deficit-adjusted control approach,” *Proc 2013 IEEE Int. Conf Rob Auto (ICRA)*, pp. 2175-82, 2013.
- [38] K. Lohse, N. Shirzad, A. Verster, N. Hodges, and H. M. Van der Loos), “Video games and rehabilitation: using design principles to enhance engagement in physical therapy,” *Journal of Neurologic Physical Therapy*, vol. 37(4), pp. 166-75, 2013.
- [39] J. W. Burke, M. D. J. McNeill, D. K. Charles, P. J. Morrow, J. H. Crosbie, and S. M. McDonough, “Optimising engagement for stroke rehabilitation using serious games,” *The Visual Computer*, vol. 25(12), pp. 1085, 2009.

- [40] P. Rego, P. M. Moreira, and L. P. Reis, June, "Serious games for rehabilitation: A survey and a classification towards a taxonomy," *Proc. of IEEE Information Systems and Technologies Iberian Conf.*, pp.1-6, 2010
- [41] S. T. Smith, and D. Schoene, "The use of exercise-based videogames for training and rehabilitation of physical function in older adults: current practice and guidelines for future research," *Aging Health*, vol. 8(3), pp. 243-52, 2012.
- [42] H. Sveistrup, "Motor rehabilitation using virtual reality," *Journal of Neuroengineering and Rehabilitation*, vol. 1(1), pp. 10, 2004
- [43] S. Saleh, G. Fluet, Q. Qiu, A. Merians, S. V. Adamovich., and E. Tunik, "Neural Patterns of Reorganization after Intensive Robot-Assisted Virtual Reality Therapy and Repetitive Task Practice in Patients with Chronic Stroke," *Frontiers in Neurology*, vol. 8, pp. 452, 2017.
- [44] M. K. Holden, and T. Dyar, "Virtual environment training: a new tool for neurorehabilitation," *Journal of Neurologic Physical Therapy*, vol. 26(2), pp. 62-71, 2002
- [45] J. Fung, C. L. Richards, F. Malouin, B. J. McFadyen, and A. Lamontagne, "A treadmill and motion coupled virtual reality system for gait training post-stroke," *CyberPsychology & Behavior*, vol. 9(2), pp. 157-62, 2006.
- [46] K. Brüttsch, T. Schuler, A. Koenig, L. Zimmerli, S. Mérillat, L. Lünenburger, and A. Meyer-Heim, "Influence of virtual reality soccer game on walking performance in robotic assisted gait training for children;" *Journal of Neuroengineering and Rehabilitation*, vol. 7(1), pp. 15, 2010.

- [47] M. Ma, M. McNeill, D. Charles D, S. McDonough, J. Crosbie, L. Oliver, and C. McGoldrick, "Adaptive virtual reality games for rehabilitation of motor disorders," *Universal Access in Human-Computer Interaction. Ambient Interaction*, pp. 681-90, 2007
- [48] R. N. Goodman, J. C. Rietschel, A. Roy, B. C. Jung, J. Diaz, R. F. Macko, and L. W. Forrester. "Increased reward in ankle robotics training enhances motor control and cortical efficiency in stroke," *J Rehabil Res Dev.*, vol. 51, pp. 213-27, 2014.
- [49] Kosiakoff. A., Sweet. W. N., Seymour. S. J., Biemer. S. M., *Systems Engineering Principles and Practice*, edition 2, 2011, p. 139.
- [50] Kosiakoff. A., Sweet. W. N., Seymour. S. J., Biemer. S. M., *Systems Engineering Principles and Practice*, edition 2, 2011, p. 149.

## Appendix -1: Requirements Traceability Matrix

The following contains the requirements traceability matrix that provides traceability from the user needs to user requirements to the necessary constraints. The index is same as listed in section 2.1.3

Requirement	User Needs	Constraints
SR 1.1.1	UN 9	-
SR 1.1.2.1	UN 9	-
	UN 7	-
SR 1.1.2.2	UN 9	-
	UN 7	-
SR 1.2.1	UN 9	-
SR 1.2.2.1	UN 9	-
	UN 7	-
SR 1.2.2.2	UN 9	-
	UN 7	-
SR 2.1.1	UN 3	SR 6.1.1
		SR 6.1
SR 2.2.1	UN 6	-
	UN 2	-
	UN 4	-
SR 2.3	UN 7	-
	UN 1	-
SR 2.4.1	UN 7	SR 6.2
		SR 6.2.1
SR 2.5	UN 4	SR 6.3
SR 3.1.1.1	UN 8	-
SR 3.1.2.1.1	UN 6	-
	UN 3	-
SR 3.1.2.1.2	UN 6	-
	UN 3	-
SR 3.1.2.1.3.1	UN 6	-
	UN 3	-
SR 3.1.2.1.3.2	UN 6	-
	UN 3	-
SR 3.1.2.1.3.3	UN 6	-
	UN 3	-
SR 3.1.2.1.3.4	UN 6	-
	UN 3	-
SR 3.1.2.1.4	UN 9	-

SR 3.1.2.2.1	UN 1	-
	UN 6	-
	UN 2	-
SR 3.1.2.2.2.1	UN 1	-
	UN 6	-
	UN 2	-
SR 3.1.2.2.2.2	UN 1	-
	UN 6	-
	UN 2	-
SR 4.1	UN 3	-
SR 4.2	UN 3	-
SR 5.1	UN 2	-
SR 5.2	UN 2	-
SR 5.3	UN 2	-
SR 5.4	UN 2	-