DESIGN OF AN INTELLIGENT CONTROL ARCHITECTURE FOR REHABILITATION ROBOTICS

By

Duygun Erol

Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Electrical Engineering

August, 2007

Nashville, Tennessee

Approved:

Professor Nilanjan Sarkar

Professor George E. Cook

Professor Mitch Wilkes

Professor Michael Goldfarb

Professor Thomas E. Groomes

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I like to express sincere appreciation to my advisor, Dr. Nilanjan Sarkar and his family. I would like thank him for his invaluable support and continual guidance, and for his encouragement and advice throughout this study. I would like thank Dr. Thomas E. Groomes, and therapist Sheila Davy for their feedback about design of the rehabilitation tasks. They helped me to visualize the work in terms of rehabilitation perspective. I would like to express my appreciation to my other committee members Dr. Mitch Wilkes, Dr. Michael Goldfarb, and Dr. George E. Cook for their helpful suggestion and comments, which guided and challenged my thinking, substantially improving this work. I gratefully acknowledge the Vanderbilt Discovery Grant that fully supported this work.

I would like thank to Dr. Kevin Fite, Robin Midgett, Tony Costi, Lewis F. Saettel and Mark Felling for their technical support throughout this study.

My deep gratitude, appreciation, indebtedness and thanks go to my dad, my mother, my sister, my brother-in-law, my aunt and my nephews. Thanks for their belief in me all the time, for their understanding and encouragement from far away. Without their boundless love and encouragement, this work could not be happened. I would like to give my very special thanks to my soul mate Utku for his invaluable support and continual encouragement through the overseas. Thanks for being patient for these years.

I would like to thank my friends Ilknur, Safak, Umut, Bilge, Canan, Filiz, Hande, Turker, Gulfem, Mert (Karakas), Mert (Tugcu), Nilgun and Huseyin in Nashville for their continuous support during my research. I like to also thank my colleagues Karla, Pramila, Bibhrajit, Changchun, Naim, Jadav, Furui, Yu and Vishnu at Robotics and Automations Systems Laboratory for their support.

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION AND SUMMARY

Introduction

Stroke is a highly prevalent condition, especially among the elderly, that results in high costs to the individual and society [1][2]. It is a leading cause of disability, commonly involving deficits of motor function. Every 45 seconds in the United States, someone suffers a stroke, which is an interruption in circulation to part of the brain. About 700,000 Americans will have a stroke this year, making it the nation's number 3 killers, according to the American Stroke Association. Anything that could help a patient regain useful function of a limb and help with activities of daily living, and make them more independent would be useful.

Clinical results have indicated that movement assisted therapy can have a significant beneficial impact on a large segment of the population affected by stroke. In recent years, new techniques adopting a task-oriented approach have been developed to encourage active training of the affected limb, which assume that control of movement is organized around goal-directed functional tasks [3][4]. “Shaping” is one of the task oriented behavioral training techniques employed in Constraint-Induced Movement (CI) therapy [3]-[5], which has the effect of placing optimal adaptive task practice procedures into a systematic, standardized and quantified format. The availability of such training techniques, however, is limited by the amount of costly therapist’s time they involve, and the ability of the therapist to provide controlled, quantifiable and repeatable assistance to complex arm and hand motion. Consequently, robot assisted rehabilitation could be used to automate labor-intensive training technique, to provide programmable levels of assistance to the patients, and to quantitatively monitor and adapt to the patient’s progress during rehabilitation.

Robot-Assisted Rehabilitation Systems

Robot-assisted rehabilitation for physical rehabilitation has been an active research area for the last few years to assist, enhance and quantify rehabilitation [6]- [18]. The robot-assisted therapies provide autonomous training where patients are engaged in repeated and intense practice of goal-directed tasks leading to improvements in motor function. Several rehabilitation robotic systems are being developed to automate therapy for the arm and hand following stroke. The MIT-Manus (Massachusetts Institute of Technology Manus) [6][7],

Assisted Rehabilitation and Measurement (ARM) Guide [8][9], Mirror Image Movement Enabler (MIME) [10]-[12] and GENTLE/s [13] are developed to facilitate arm movement of the stroke patients. New rehabilitation therapy environments are under development to permit the training of real-life functional tasks involving reaching and grasping [14]. Rutgers Master II-ND [15], a hand exoskeleton [16], the CyberGrasp [17], and a robotic device (HWARD) [18] are also being developed to facilitate the hand movement of the stroke patients.

MIT Manus (Figure 1) is the pioneering robotic rehabilitation system which was developed at the Massachusetts Institute of Technology (MIT) in the early 1990s [6],[7]. The MIT Manus therapy robot permits stroke survivors to practice two-dimensional (2-D) point-to-point movements. MIT Manus uses impedance controller to provide assistance to move the patient's arm to the target position, where the patients can visually see their movement and the target location. It is configured for safe and stable operation in close physical contact with humans. Stroke patients have demonstrated improvements in shoulder strength and executing smooth movements by using MIT Manus. For example, the patients could follow a desired circular trajectory more closely with training with MIT Manus and their velocity profile began to look like a bell-shape with less corrective movements.

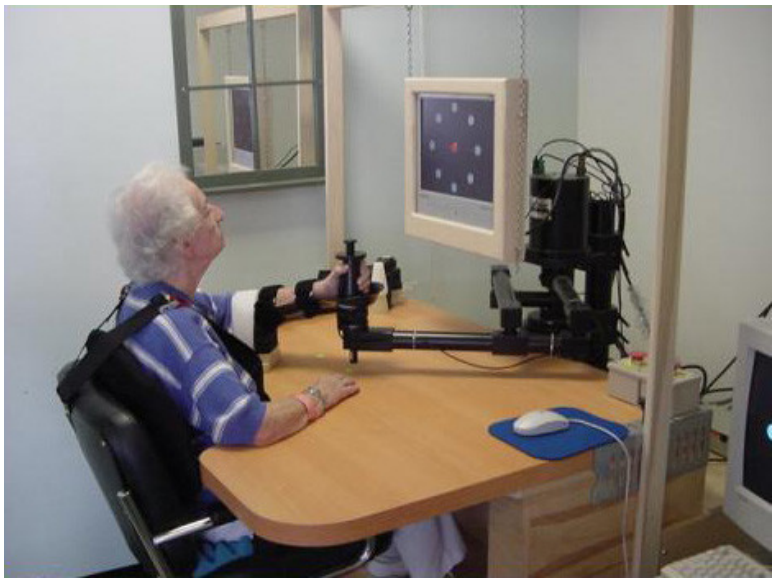


Figure 1. MIT Manus

The other well-known robot-assisted rehabilitation device called Assisted Rehabilitation and Measurement (ARM) Guide [8],[9] is developed at the Rehabilitation Institute of Chicago. ARM Guide is developed to allow stroke subjects to perform the reaching tasks, which is capable of generating both horizontal and vertical motion (Figure 2). The subject's

forearm/hand is strapped to a specially designed splint that slides along a linear constraint (Figure 2). A motor assists or resists the arm movement along the linear bearing. The device is mounted on a stand for height adjustment, and can be flipped to measure reaching with the left/right hand. The orientation of the ARM Guide can be manually changed for either vertical or horizontal planes, however the device, can not provide assistance to the patients for both planes at the same time. A guided-force training algorithm is designed with the ARM Guide to train the patients. In this training algorithm the patients attempt to reach forward at a comfortable speed to the end of his/her range of motion. Throughout the movement, a six-axis load cell is monitored for off-axis forces above a 10 N threshold. Once such a force is detected, the motor locks the position of the hand piece and the user receives real-time graphical feedback of the force error. When the six-axis load cell detects appropriate forces toward the target, the graphical cue is removed and the motor unlocks the hand piece, which allows the user to progress toward the target. When the subject is moving toward the task, the ARM Guide resists/assists the subject's movement. A graphical interface on a computer monitor represents the current position of the ARM Guide and a target position.

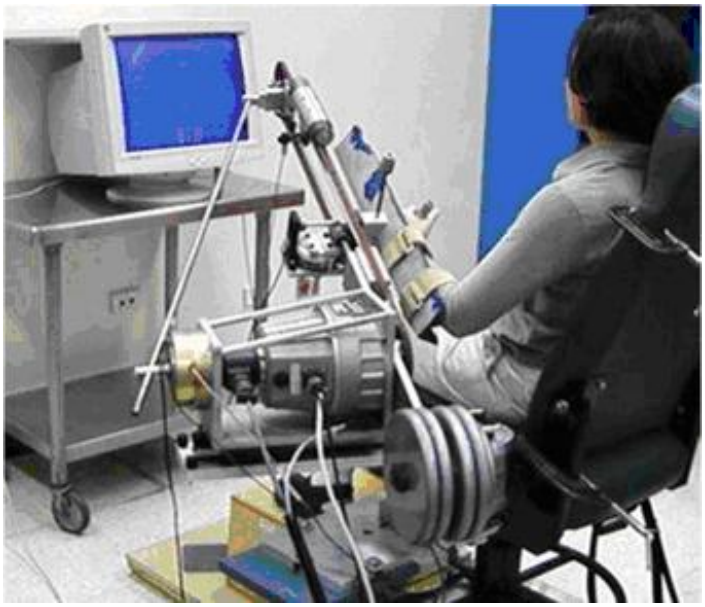


Figure 2. Assisted Rehabilitation and Measurement Guide (ARM Guide)

Mirror Image Movement Enabler (MIME) robotic rehabilitation device is developed at Stanford University and the Veterans Affairs Palo Alto Health Care System [10]-[12]. MIME robot permits stroke survivors to practice three-dimensional (3D) point-to-point reaching movements occurring in the real world. A PUMA-560 robot is mounted beside the table, which is modified so that the subjects can interact with the robot in a stable and repeatable

manner (Figure 3). The subject's arm is strapped into the splint with the wrist. The interaction force between the subject and the device during the reaching tasks are measured using a force sensor. A quick-release coupling mechanism is designed to shut down power when a certain torque level is exceeded. The motion of the robot is monitored in order to prevent potentially hazardous situations from occurring. Several modes of the robot-assisted movement have been implemented with MIME, including passive, active-assisted, and active-constrained, as well as a bimanual mode in which MIME continuously moves the impaired limb to the mirror image position of the unimpaired limb as measured with a digitizing linkage. In the passive mode, the patient remains passive while the robot moves the arm along a pre-programmed position trajectory using proportional-integral-derivative (PID) controller. In the active assisted mode the patient initiates the movement and the robot assists and guides the motion along the desired position trajectory. In the active-constrained mode, a force sensor attached to a PUMA 560 robotic manipulator which measures the direction of the force generated by the patient’s hand at the interface between the hand and the robot. If the patient applies force in the desired direction, then the robot moves the subject’s arm in the direction of the motion. If the applied force is misdirected, then the robot stops moving toward the target and programmed impedance allowed the robot to deflect slightly in the direction of the force. Subjects have improvement in a scale that measures functional independence in Activities of Daily Living (ADL). Patients who received robot therapy had statistically larger improvements in the motor impairment scale and larger gains in strength and reach extent.

Figure 3. Mirror Image Movement Enabler (MIME)

GENTLE/s is developed at the University of Reading for neurorehabilitation of stroke patients. GENTLE/s is designed to both improve the quality of treatment and reduce costs (Figure 4) [13]. GENTLE/s permits stroke survivors to practice three-dimensional (3D) point-to-point reaching movements occurring in a haptic virtual environment. A 3 Degree-of-freedom (DOF) robot called HapticMaster is integrated with a virtual environment. Subjects practice “reach-and-grasp” type of movements (without the grasp component) through interaction with a virtual room. The patient’s arm is put in an elbow orthosis with wires suspending it from an overhead frame in order to eliminate the gravity effect. The GENTLE/s provides the assistance to the patients to move to the target positions along with a predefined path using admittance control. The visual guidance is provided in the form of start/end points for a specific movement pattern. There are 3 different levels of movement control, which are passive, active-assisted and active mode. In the patient passive mode, the patient lacks the power to initiate the movement and remains passive, and the device moves the arm along the pre-defined path. In active-assisted mode, the device starts moving as long as the patient initiates the movement and the device helps the user to reach the desired position. In active mode, unlimited time is given to the patients to finish the correct task and the device provides assistance when the patient deviates from the pre-defined path. The results show that the GENTLE/s is effective in improving recovery and the device has been accepted from both the subjects and the therapists. Similar robotic rehabilitation system has been used in Medical College of Wisconsin, which is called the Activities of Daily Living Exercise Robot (ADLER), for training of the stroke patients (Fig. 5) [14]. ADLER supports seated functional tasks such as drinking, eating and game playing tasks like tic-tac-toe. A Functional Electrical Stimulation (FES) glove for grasping action is under development to be integrated to ADLER system to perform the ADL tasks.

Figure 4. GENTLE/s

Figure 5. Activities of Daily Living Exercise Robot *(*ADLER)

A similar number of hand assistive devices have also been reported in the literature. Rutgers Master II-ND [15] is one of the hand assistive devices designed for dextrous interactions with virtual environments. Rutgers Master II-ND uses a unique design to actuate the tips of three fingers as well as the thumb (Fig. 6). The device is developed in the State University of New Jersey. The Rutgers Master II-ND uses custom-made pneumatic cylinders to push the fingertips out from the palm. In order to determine the angle of the pneumatic cylinder two hall-effect sensors are placed at the base of each pneumatic cylinder. Additionally two infrared sensors inside the cylinders are used to measure the travel distance of the piston. The angle and cylinder length is used to determine the position of the fingertip. Rutgers Master II-ND is an example of a lightweight and palm-mounted finger manipulator. It is still not clear whether it is possible to pull the fingers in toward the palm.

Figure 6. The Rutgers Master II-ND

Another hand device, which is capable of actuating all three joints of the index finger in both flexion and extension, is developed by a team at Carnegie Mellon University [16] (Fig. 7). The Carnegie Mellon University hand device is developed as a manipulator to assist the patients for their grasping activity. Two pneumatic cylinders are used to flex and three springs are used to extend these three joints of the index. However, this hand device is not capable of actuating the proximal interphalangeal (PIP) and the distal interphalangeal (DIP) joints independently. Motion of the DIP and the PIP joints are coupled. Electromyography (EMG) signals from the patient's forearm are used to determine the intention of the patients during the grasping activity.

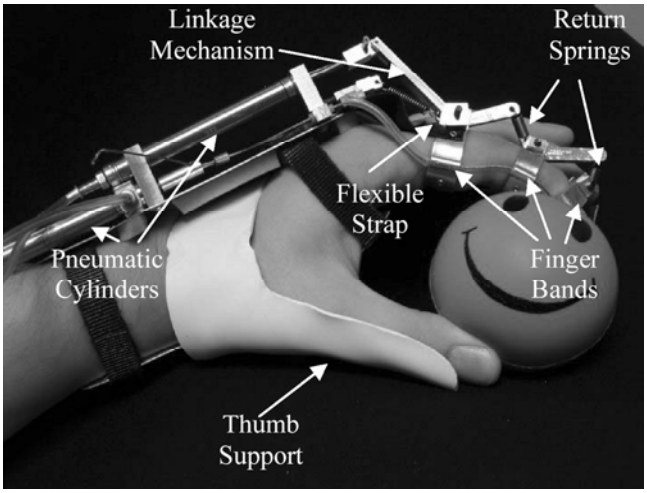


Figure 7. The Carnegie Mellon Hand Device

CyberGrasp [17] is the only commercially available hand assistive device which is built to provide extensive forces to the tips of the fingers and the thumb for grasping. However, this device cannot provide flexive forces (Fig. 8). The CyberGrasp is developed as part of a CyberGlove for interactions with virtual environments, and has been successfully used in medical applications and remote handling of hazardous materials.

Figure 8. CyberGrasp

A Hand-Wrist Assisting Robotic Device (HWARD) as shown in Fig. 9 has been developed to assist repetitive grasping and releasing movements while allowing the patient to feel real objects during therapy [18]. HWARD is a 3 degrees-of-freedom (DOF) which is pneumatically-actuated backdriveable robotic device.

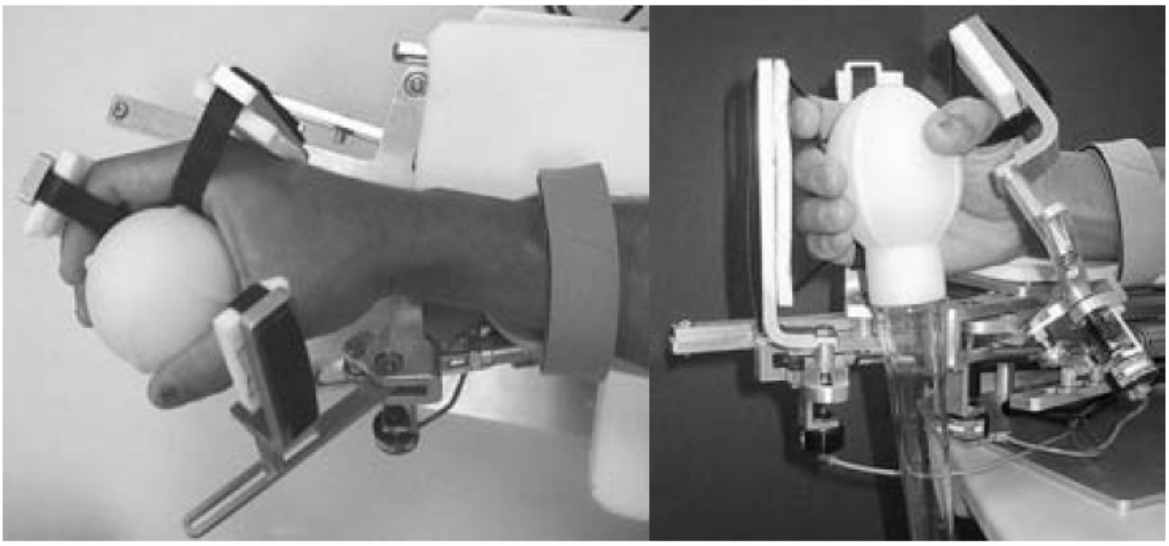


Figure 9. Hand-Wrist Assisting Robotic Device (HWARD)

Current State-of-the-Art in Robot-Assisted Rehabilitation

There are significant research activities in the development of new methodologies for robot-assisted rehabilitation in the last few years. The promising results of the above-mentioned rehabilitation robotic systems indicate that the robots could be used as effective rehabilitation tools. Current theories of stroke rehabilitation point towards paradigms of intense and repetitive use of the affected limb as a means for motor program reorganization.

However, it has also been demonstrated in [19] that repetitive execution of simple motor tasks may not be as effective as execution of more complex motor tasks that involve in-depth cognitive processing. Thus, it would be useful if a rehabilitation task can be designed, where the patients not only make repetitive movement but also pay attention to tracking accuracy which requires cognitive processing. However, in such a tracking task, the patients may not be able to track the desired motion because of their impairments. Thus a low-level assistive controller is required to provide assistance to the patient to complete this tracking task in an accurate manner.

It has also been noticed that the existing robotic rehabilitation systems primarily use low-level assistive controllers to assist the movement of the patients' arms. For example, MIT Manus uses an impedance controller [6],[7], MIME uses a PID controller [10]-[12] and GENTLE/s uses an admittance controller [13] for arm movement assistance. However, to our knowledge, none of these systems has a dedicated high-level decision making mechanism that can comprehensively monitor the task, provide assessment of the progress, and alter the task parameters to impart effective therapy based on the patient's performance in an automated manner. In these cases, therapist decides the task parameters (e.g., desired reaching distance, desired speed of motion etc.) and continuously monitors the patient's progress to update the task parameters and to ensure the safety of the patients. As a result, it is likely to consume more time of the therapist, increase the workload of the therapist, and consequently, increase the cost of treatment. Thus, in this case a high-level decision making mechanism will likely to help in therapist's decision-making and make the robotic rehabilitation process safer.

Additionally, in robot-assisted therapies the patients attempt to make voluntary movements while the robotic device provides assistance to complete the desired rehabilitation task. Many of these existing robotic rehabilitation devices ensure the smoothness of the movement by specifying the desired trajectory of the task as the minimum jerk trajectory as originally proposed by [28]. The idea is that if the subject can follow the desired minimum jerk trajectory, the motion will be smooth. However, the ability of the subject to follow a prescribed trajectory and therefore result in smooth interaction is a function of the interaction dynamics between the robot and the subject. Thus, it could be useful if a control framework can be designed to modify the interaction dynamics between the robot and the subject which will result in smooth interaction.

Moreover, it has been noticed that the existing arm and hand rehabilitation systems are limited by their inability to simultaneously assist both arm and hand movements. This

limitation is critical because the stroke therapy literature supports the idea that the ADL focused tasks (emphasis on task-oriented training), which engage patients to perform the tasks in enriched environments have shown significant increase in the motor recovery after stroke [39]-[42]. Robots that cannot simultaneously assist both arm and hand movements are of limited value in the task-oriented approach that involve practicing complex tasks. Thus, it is desirable to design a control framework to achieve the desired coordinated motion between the arm movement and the hand movement.

Scope and Summary of the Research

In this dissertation, we initially design a low-level assistive controller to be used to provide robotic assistance as and when needed to the subjects to complete a rehabilitation task. This rehabilitation task is designed in such a way that the subjects not only make repetitive arm movement but also pay attention to the tracking accuracy of the desired motion trajectory that requires cognitive processing. Then, the low-level assistive controller is modified to provide the robotic assistance to the subjects in such a manner that the interaction between the robot and the subject is smooth. Later, a control architecture is presented in terms of a hybrid system model combining a high-level controller and the low-level assistive controller. The high-level controller is designed to help the therapist in: i) determining the task parameters dynamically based on the patients' performance and implementing the new set of parameters; and ii) monitoring the safety related events in an automated manner and generating an accommodating plan of action should such an event happen. Finally, the control architecture has been augmented to coordinate multiple assistive devices (arm and hand assistive devices) which will enable the subjects to perform the ADL tasks. As a part of this work, a rehabilitation robotic system is built in the laboratory. The real-time experimental setup and rehabilitation tasks are designed with the consultation of a physical therapist to demonstrate the efficacy of the presented work. The proposed research is presented in 4 manuscripts which are given as:

Manuscript 1: Design and Implementation of an Assistive Controller for Rehabilitation Robotic Systems

Background

Current theories of stroke rehabilitation focuses on the intense and repetitive use of the affected limb. However, it has also been demonstrated in [19] that repetitive execution of

simple motor tasks may not be as effective as execution of more complex motor tasks that involve in-depth cognitive processing. Precision-demanding tasks that challenge motor learning processes create richer conditions for change in the brain reorganization on rats [20][21], primates [22][23] and human [24]. It was shown that the movement tracking training that requires cognitive processing achieved greater gains in performance than that of movement training that did not require cognitive processing [19][25]. Additionally, it was shown that finger movement tracking training produced greater gains in the range of motion and tracking accuracy compared to finger movement training that required no temporospatial processing [26].

Summary of Contribution

There are two contributions in this work, first is to design a rehabilitation task such that the subjects not only make repetitive movements but also pay attention to the tracking accuracy of the desired motion trajectory, and the second is to design a controller to provide the robotic assistance to the subjects to help them to track the rehabilitation task in an accurate and concentrated manner. The effectiveness of the method is verified by the real-time experiments on unimpaired subjects. Manuscript 1 is based on the following paper:

- Erol D. and Sarkar N., "*Design and Implementation of an Assistive Controller for Rehabilitation Robotic Systems*", International Journal of Advanced Robotic Systems (accepted to be published in vol. 4 no.3, September 2007).

Manuscript 2: Intelligent Control for Robotic Rehabilitation after Stroke

Background

It has been noticed that the existing robotic rehabilitation systems mainly use low-level assistive controllers to assist the movement of the patients' arms [6],[7], [10]-[13]. In these cases, the task parameters (e.g., desired reaching distance, desired speed of motion etc.) are pre-defined at the beginning of the therapy and the therapist continuously monitors the patient's progress and quickly determines if the task parameters are needed to be changed to adapt to the patients' performance. Then these task parameters are updated by the therapist to be executed by the low-level assistive controllers. In addition, the therapist is required to pay attention to the safety related issues during the robot assisted therapy. If such an issue occurs, the therapist needs to take necessary action to ensure the safety of the patients.

Summary of Contribution

The main contribution of this work is design of an intelligent controller, which is called a high-level controller, i) to monitor the progress of the patient to determine the task parameters dynamically; and ii) to monitor safety related events in an automated manner to make decisions on the modification of the task that might be needed for the therapy. We believe that such an intelligent controller will likely to help in therapist's decision-making and make the robotic rehabilitation process safer. Manuscript 2 is based on the following papers:

- Erol D. and Sarkar N., "*Intelligent Control for Robotic Rehabilitation after Stroke",* Journal of Intelligent and Robotic Systems (under review).
- Erol D., Sarkar. N and Halder B., "*A High-Level Controller for Robot-Assisted Rehabilitation,*" IEEE International Conference of the Engineering in Medicine and Biology Society (EMBS), New York, USA, 2006, pp. 3234 – 3237.
- Erol D. and Sarkar. N, "*Intelligent Control Framework for Robotic Rehabilitation after Stroke*", IEEE International Conference on Robotics and Automation, (ICRA), Rome, Italy, 2007, pp. 1238-1243.

Manuscript 3: An Approach to Smooth Human-Robot Interaction in Robot-Assisted Rehabilitation

Background

In robot-assisted therapies, the patients attempt to make voluntary movements while the robotic device provides assistance to complete the desired rehabilitation task. It is desirable to provide the robotic assistance to the patients in such a manner that the resulting interaction between the robot and the patient is smooth. In the literature jerk has been used as a measure of the smoothness of the prescribed motion, where minimum jerk implied smooth movement [27],[29]. The numerical value of the jerk has been used as a metric to determine the movement smoothness during stroke recovery [30]. Furthermore, smoothness of the movement has been quantified by the mean squared magnitude of the jerk [27],[29],[31]. Many of the existing robotic rehabilitation devices such as [8]-[13] try to ensure the smoothness of the motion by specifying the desired trajectory of the task as the minimum jerk trajectory as originally proposed by [28]. The idea is that if the subject can follow the desired minimum jerk trajectory, the motion will be smooth and consequently, the force applied by

the subject will also be smooth. Here the smoothness is measured by the rate of the change of the force applied by the subject. However, when the subject cannot follow the specified desired motion trajectory entirely by his/her own effort, he/she will need robotic assistance. Note that the manner in which this robotic assistance is imparted could affect the rate of change of the subject applied force. For example, if the robot provides assistance with a high overshoot, then the subject is required to overcome this overshoot by changing his/her applied force in a rapid manner. This scenario could result in more variation in subject applied force, which implies non-smooth interaction.

Summary of Contribution

The main contribution of this work is to modify the low-level assistive controller in such a way that the interaction between the robot and the subject is smooth. We have shown that suitable modification of the interaction dynamics between the robot and the subject will result in less variation of the force applied by the subject (smooth interaction) to complete the task for any type of desired trajectory including the minimum jerk trajectory. A control framework is proposed in here, which is called an artificial neural network (ANN) based Proportional-Integral (PI) gain scheduling controller, that will automatically adjust the control gains for each subject such that the resultant interaction dynamics between the subject and the robot could result in smooth interaction. The control gains are determined based on online estimation of the human arm parameters (i.e., stiffness). The ANN, which is trained offline, adjusts the control gains online. The proposed controller combines the benefit of system identification technique with the robustness of neural network-based methods. Manuscript 3 is based on the following papers:

- Erol D and Sarkar. N, "*An Approach to Smooth Robotic Assistance for Rehabilitation*", International Journal of Robotics Research (under review).
- Erol D and Sarkar. N, "*Smooth Human-Robot Interaction in Robot-Assisted Rehabilitation*", IEEE International Conference on Rehabilitation Robotics: Frontiers of the Human-Machine Interface (ICORR), Noordwijk, Netherlands, 2007 (accepted).
- Erol D., Mallapragada V., Sarkar N, Uswatte G. and Taub E.,"*A New Control Approach for Robot Assisted Rehabilitation*", IEEE International Conference on Rehabilitation Robotics: Frontiers of the Human-Machine Interface (ICORR), Chicago, USA, 2005, pp. 323 - 328.

- Erol D., Mallapragada V., Sarkar N, Uswatte G. and Taub E., "*Autonomously Adapting Robotic Assistance for Rehabilitation Therapy*", IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob), Pisa, Italy, 2006, pp. 567 – 572.
- Erol D., Mallapragada V. and Sarkar N., "*Adaptable Force Control in Robotic Rehabilitation*", IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN), Nashville, USA, 2005, pp. 649 - 654.
- Mallapragada V. Erol D. and Sarkar N., "*A New Method of Force Control for Unknown Environments, Advanced Robotic*", International Journal of Advanced Robotic Systems (accepted to be published in vol. 4 no.3, September 2007).
- Mallapragada V., Erol D. and Sarkar N., "*A New Method for Force Control for Unknown Environments*", IEEE International Conference on Intelligent Robots and Systems, (IROS), Beijing, China, 2006, pp. 4509 – 4514.

Manuscript 4: Coordinated Robotic Assistance for Activities of Daily Living Tasks

Background

Even though existing arm and hand rehabilitation systems have shown promise of clinical utility, they are limited by their inability to simultaneously assist both arm and hand movements. This limitation is critical since ADL focused tasks have shown to increase in the motor recovery after stroke [39]-[42]. The great majority of the activities of daily living (ADL) tasks require participation of both the arm and the hand movements..

Summary of Contribution

The main contribution of this paper is to present a new control architecture that coordinates assistive devices in a systematic manner as necessary for a particular ADL task. This control architecture exploits hybrid system modeling techniques to provide robotic assistance for ADL tasks. Hybrid control framework has been effectively used in other fields, such as industrial robotics, medicine, and manufacturing [38], however, it is applicability for rehabilitation purposes is new. We argue that the proposed control architecture based on hybrid system framework could be useful in coordinating the assistive device controllers in a safe and complex manner to satisfy a variety of ADL task requirements. Such a controller is expected to address the need in the field of the rehabilitation robotics. Manuscript 4 is based on the following paper:

- Erol D. and Sarkar N., "*Coordinated Robotic Assistance for Activities of Daily Living*", IEEE International Conference on Intelligent Robots and Systems, (IROS), San Diego, USA 2007 (Submitted).
- Erol D. and Sarkar N., "*Supervisory Control Architecture for Coordination of Arm and Hand Robotic Assistive Devices for Activities of Daily Living Tasks",* IEEE Transactions on Neural and Rehabilitation Engineering (Preparation for Submission).

References

[1] American Heart Association: Heart and Stroke Statistical Update, Available from: http://www.Americanheart.org/statistics/stroke.htm, 2006.

[2] Matchar D.B., Duncan P.W. Cost of stroke. Stroke Clin Updates 1994; 5:9-12.

[3] E. Taub, N.E. Miller, T.A. Novack, E.W. III Cook, W.C. Fleming, C.S. Nepomuceno, J. S. Connell, J.E. Crago, "Technique to improve chronic motor deficit after stroke", Arch Phys Med Rehabil, 74, pp. 347-354, 1993.

[4] B. T. Volpe, "Stroke, Stroke": A coxswain's call for more work and more innovation", J. of Rehab. Research Development, 41:3A, 2004.

[5] E. Taub, G. Uswatte, R. Pidikiti, "Constraint-Induced Movement Therapy: A new family of techniques with broad application to physical rehabilitation - a clinical review, J. of Rehab. Research & Development 36, pp.237-251, 1999.

[6] Krebs, H.I.; Palazzolo, J.J.; Dipietro, L.; Ferraro, M.; Krol J.; Rannekleiv, K.; Volpe, B.T. & Hogan N. Rehabilitation Robotics: Performance-Based Progressive Robot-Assisted Therapy. Autonomous Robots, 15, 1, pp. 7-20, 2003.

[7] Krebs, H. I.; Ferraro, M., Buerger, S.P., Newbery, M. J., Makiyama, A., Sandmann, M.; Lynch, D.; Volpe, B. T. & Hogan, N. Rehabilitation robotics: pilot trial of a spatial extension for MIT-Manus. Journal of NeuroEngineering and Rehabilitation. 1, 5, pp. 1-15, 2004.

[8] Kahn, L.E.; Zygman, M.L; Rymer, W.Z. & Reinkensmeyer, D.J. Robot-assisted reaching exercise promotes arm movement recovery in chronic hemiparetic stroke: a randomized controlled pilot study. Journal of NeuroEngineering and Rehabilitation, 3, 12, pp. 1-13, 2006.

[9] Kahn L.E; Lum P.S.; Rymer W.Z; Reinkensmeyer D.J. Robot-assisted movement training for the stroke-impaired arm: Does it matter what the robot does? J. of Rehab. Research & Development, 43, 5, pp. 619-630, 2006.

[10] Lum, P.S.; Burgar, C.G.; Kenney, D.E. & Van der Loos H.F.M. Quantification of force abnormalities during passive and active-assisted upper-limb reaching movements in

post-stroke hemiparesis. IEEE Transactions Biomedical Engineering. 46, 6, pp. 652-661, 1999.

[11] Burgar CG, Lum PS, Shor, Van der Loos HFM: Development of robots for rehabilitation therapy: The Palo Alto VA/Stanford experience. Journal of Rehabilitation Research, and Development November/December 2000; 37(6).

[12] Lum, P.S.; Burgar, C.G.; Van der Loos, H.F.M.; Shor, P.C.; Majmundar M.; Yap R. MIME robotic device for upper-limb neurorehabilitation in subacute stroke subjects: A follow-up study . J. of Rehab. Research & Development, 43,5, pp. 631-642, 2006.

[13] Loureiro, R.; Amirabdollahian, F.; Topping, M.; Driessen, B. & Harwin, W. Upper limb mediated stroke therapy - GENTLE/s approach. Autonomous Robots. 15, pp. 35-51, 2003.

[14] M. J. Johnson, K. J. Wisneski, J. Anderson, D. Nathan, and R. Smith, "Development of ADLER: The Activities of Daily Living Exercise Robot," IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, Pisa, Italy, 2006, pp. 254-259.

[15] D. Jack, R. Boian, A. S. Merians, M. Tremaine, G. C. Burdea, S. V. Adamovich, M. Recce, and H. Poizner, "Virtual reality-enhanced stroke rehabilitation," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 9, pp. 308–318, 2001.

[16] M. DiCicco, L. Lucas, and Y. Matsuoka, "Comparison of control strategies for an EMG controlled orthotic exoskeleton for the hand," in Proc. of IEEE International Conference on Robotics and Automation, 2004, pp. 1622 – 1627.

[17] Immersion Corporation, http://www.immersion.com/.

[18] C. D. Takahashi, L. Der-Yeghiaian, V. H. Le, and S. C. Cramer, "A robotic device for hand motor therapy after stroke" in IEEE 9th International Conference on Rehabilitation Robotics, 2005, pp. 17 – 20.

[19] Carey, J.R.; Bhatt, E. & Nagpal, A. Neuroplasticity Promoted by Task Complexity. Exercise and Sport Science Review, 33, pp. 24-31, 2005.

[20] Black, J.E.; Isaacs, K.R.; Anderson, B.J.; Alcantara, A.A. & Greenough, W. T. Learning causes synaptogenesis, whereas motor activity causes angiogenesis, in cerebellar cortex of adult rats. Proceedings of National Academic. Science, pp. 5568-5572, 87(15), USA, 1990.

[21] Kleim, J. A.; Barbay, S. & Cooper, N.R. Motor learning-dependent synaptogenesis is localized to functionally reorganized motor cortex. Neurobio Lean Mem, 77, pp. 63-77, 2002.

[22] Plautz, E.J.; Milliken, G. W. & Nudo, R J. Effects of repetitive motor training on movement representations in adult squirrel monkeys: role of use versus learning. Neurobiol. Learn. Mem., 74, pp. 27-55, 2000.

[23] Nudo, R.; Milliken, G.; Jenkins, W. & Merzenich M. Use-dependent alterations of movement representations in primary motor cortex of adult squirrel monkeys. J. Neurosci, 16, 2, pp. 785-807, 1996.

[24] Pascual-Leone, A.; Nguyen, K.T.; Kohen, A.D.; Brasil-Neto, J.; Cammarota, A. & Hallett M. Modulation of muscle responses evoked by transcranial magnetic stimulation during the acquisition of new fine motor skills. J. Neurophysiol. 74, pp. 1037-1045, 1995.

[25] Carey, J.R.; Anderson, K.M.; Kimberley, T.J., Lewis, S.M.; Auerbach E.J. & Ugurbil, K. fMRI anaysis of ankle movement tracking training in Participants with Stroke. Exp Brain Res, 154, pp. 281-290, 2004.

[26] Carey J.; Durfee, W.; Bhatt, E. ; Nagpal, A.;Weinstein, S.; Anderson, K. & Lewis, S. Tracking vs. movement Telerehabilitation training to change hand function and brain reorganization in stroke.Submitted to Neurorehabilitation and Neural Repair, 2006.

[27] Flash T., Hogan N. The coordination of arm movements: an experimentally confirmed mathematical model. J Neurosci, Vol. 5, pp. 1688—1703, 1985.

[28] Hogan N., Flash T. Moving gracefully: quantitative theories of motor coordination, Trends Neurosci, Vol. 10, pp.170-174, 1987.

[29] Hogan N. 1988. Planning and execution of multijoint movements, Can J Physiol Pharmacol, Vol. 66, pp. 508-517, 1988.

[30] Rohrer B. ,Fasoli S. , Krebs H.I. ,Hughes R. ,Volpe, Frontera W.R. , Stein J., Hogan N. Movement Smoothness Changes during Stroke Recovery. The Journal of Neuroscience, 22(18):8297–8304, 2002.

[31] Stein R.B., Cody F.W., Capaday C. The trajectory of human wrist movement. J Neurophysiol, 59:1814-1830, 1988.

[32] Johnson M. J., Wisneski K. J., Anderson J., Nathan D., & Smith R., "Development of ADLER: The Activities of Daily Living Exercise Robot," IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, Pisa, Italy, 254-259, 2006.

[33] Jack D., Boian R., Merians A. S., Tremaine M., Burdea G.C., Adamovich S.V., Recce M.,& Poizner H., "Virtual reality-enhanced stroke rehabilitation," IEEE Transactions on Neural Systems and Rehabilitation Engineering, 9:308–318, 2001.

[34] DiCicco M., Lucas L., & Matsuoka Y., "Comparison of control strategies for an EMG controlled orthotic exoskeleton for the hand," in Proc. of IEEE International Conference on Robotics and Automation, pp. 1622 – 1627, 2004.

[35] Immersion Corporation, http://www.immersion.com/.

[36] Kline T., Kamper D., & Schmit B., "Control system for pneumatically controlled glove to assist in grasp activities," in IEEE 9th International Conference on Rehabilitation Robotics, pp. 78 – 81, 2005.

[37] Takahashi C. D., Der-Yeghiaian L., Le V. H., & Cramer S. C., "A robotic device for hand motor therapy after stroke" in IEEE 9th International Conference on Rehabilitation Robotics, pp. 17 – 20, 2005.

[38] Antsaklis P. J., "A Brief Introduction to the Theory and Applications of Hybrid Systems," Proceedings of the IEEE on Special Issue on Hybrid Systems: Theory and Applications, (88) 879-887, 2000.

[39] J. Carr and R. A. Shepherd, Motor Relearning Programme For Stroke, 2nd ed.: Aspen Publishers, 1987.

[40] E. Taub, J. E. Crago, and L. D. Burgio, "An operant approach to rehabilitation rehabilitation medicine: overcoming learned nonuse by shaping," Journal of Experiment Analysis of Behavior, vol. 61, pp. 281-293, 1994.

[41] J. Liepert, H. Bauder, W. H. R. Miltner, E. Taub, and C. Weiller, "Treatment-induced massive cortical reorganization after stroke in humans," Stroke, vol. 31, pp. 1210-1216, 2000.

[42] G. F. Wittenberg, R. Chen, K. Ishii, K. O. Bushara, S. Eckloff, E. Croarkin, E. Taub, L. H. Gerber, M. Hallett, and L. G. Cohen, "Constraint-induced therapy in stroke: magnetic-stimulation motor maps and cerebral activation," Neurorehabilitation and Neural Repair, vol. 17, pp. 48-57, 2003.

CHAPTER II: MANUSCRIPT 1

DESIGN AND IMPLEMENTATION OF AN ASSISTIVE CONTROLLER FOR REHABILITATION ROBOTIC SYSTEMS

Duygun Erol & Nilanjan Sarkar

(Accepted in International Journal of Advanced Robotic Systems)

Abstract

The goal of our research is to develop an assistive controller for robotic rehabilitation of the upper extremity after stroke. The controller is used to provide robotic assistance to participants to help them to track a desired motion trajectory required for the rehabilitation task in an accurate and concentrated manner. This rehabilitation task is designed to ensure concentrated repetitive motion that requires cognitive processing. Experimental results on unimpaired participants are presented to demonstrate the effectiveness and feasibility of the proposed controller.

Keywords: assistive controller, movement tracking training, robot-assisted rehabilitation

1. Introduction

Stroke is a highly prevalent condition especially among the elderly that results in high costs to the individual and society (Matchar, D.B. & Duncan, P.W., 1994). According to the American Heart Association, in the U.S., approximately 700,000 people suffer a first or recurrent stroke each year (American Heart Association, 2006). It is a leading cause of disability, commonly involving deficits of motor function.

Recent clinical results have indicated that movement assisted therapy can have a significant beneficial impact on a large segment of the population affected by stroke or other motor deficit disorders. Experimental evidence suggests that intensive movement training of new motor tasks is required to induce long-term brain plasticity. The availability of movement training techniques, however, is limited by the amount of costly therapist's time they involve and the ability of the therapist to provide controlled, quantifiable and repeatable assistance to arm movement. Consequently, robot assisted rehabilitation that can

quantitatively monitor and adapt to patient's progress, ensure consistency during rehabilitation may provide a solution to these problems.

In the last few years, robot-assisted rehabilitation for physical rehabilitation of the stroke patients has been an active research area to assist, monitor, and quantify rehabilitation therapies (Krebs 2004, Lum 2006, Kahn 2006, Loureiro 2003). These robotic devices are used to recover arm movement after stroke, which provide opportunities for repetitive movement exercise and more standardized delivery of therapy with the potential of enhancing quantification of the therapeutic process. The first robotic assistive device used as a therapeutic tool, the MIT Manus (Krebs 2003, 2004) uses impedance controller to provide assistance to move the patient's arm to the target position in an active assisted mode, where patients can visually see their movement and target location. In (Krebs 2004) they expand the capabilities of MIT Manus to include motion in a three-dimensional workspace to rehabilitate other muscle groups and limb segments than shoulder and elbow. The Mirror Image Movement Enabler (MIME) and the Assisted Rehabilitation and Measurement (ARM) Guide, expanded the investigations of therapeutic applications of robots into the chronic stroke population. MIME uses a PUMA 560 manipulator to provide assistance to move the subject's arm with a pre-programmed position trajectory using Proportional-Integral-Derivative (PID) controller (Lum 2006). ARM Guide is capable of generating both horizontal and vertical motion, and giving resistance and support to the patient (Kahn 2006). The GENTLE/s (Loureiro 2003) is a haptic robot used to provide assistance to patients to move to the target positions along with a predefined path using admittance control. The subject's movement trajectory is represented in the virtual environment.

The promising results of the above-mentioned rehabilitation robotic systems indicate that robots could be used as effective rehabilitation tools. Current theories of stroke rehabilitation point towards paradigms of intense and repetitive use of the affected limb as a means for motor program reorganization. However, it has also been demonstrated in (Carey 2005) that repetitive execution of simple motor tasks may not be as effective as execution of more complex motor tasks that involve in-depth cognitive processing. Precision-demanding tasks that challenge motor learning processes create richer conditions for change in the brain reorganization on rats (Black 1990, Kleim 2002), primates (Plautz 2000, Nudo 1996) and human (Pascual-Leone 1995). It was shown that movement tracking training that requires cognitive processing achieved greater gains in performance than that of movement training that did not require cognitive processing (Carey 2005). Additionally, it was shown that finger movement tracking training produced greater gains in the range of motion and tracking

accuracy compared to finger movement training that required no temporospatial processing (Carey 2006). Thus, it would be useful if a tracking movement training method can be developed, where the patients not only make repetitive movement but also pay attention to tracking accuracy. However, in such a tracking task, patients may not be able to track the desired motion because of their impairments. Thus, a rehabilitation robotic system can be designed to provide assistance to the patient to track the desired motion accurately based on his/her performance.

In this paper, we present a controller to be used to provide robotic assistance as and when needed to the participants to complete an upper arm rehabilitation task. This task is designed to impart movement training that requires cognitive processing. Note that the presented assistive controller is not specific to a given rehabilitation robotic system but can be integrated with any previously proposed rehabilitation systems. However, in order to demonstrate the efficacy of the proposed assistive controller, we needed to develop a rehabilitation robotic system, which is also presented in the paper.

This paper is organized as follows. It first presents the proposed rehabilitation robotic system in Section 2. The methodology section (Section 3) includes task description, controller design, and decision logic of robotic assistance. Experiments and results are presented in Section 4. Section 5 discusses the potential contributions of this work and possible future research directions.

2. The Rehabilitation Robotic System

A PUMA 560 robotic manipulator is used as the main hardware platform in this work. The manipulator is augmented with a force-torque sensor and a hand attachment device (Fig. 1).

2.1. Hardware

The PUMA 560 is a 6 degrees-of-freedom (DOF) device consisting of six revolute joints (PUMA web site). In order to record the force and torque applied by the human, an ATI Gamma force/torque sensor is used. The robot is interfaced with Matlab/Real-time Workshop to allow fast and easy system development. The force values recorded from the force/torque sensor are obtained using a National Instruments PCI-6031E data acquisition card with a sampling time of 0.001 seconds. The joint angles of the robot are measured using encoder with a sample time of 0.001 seconds from a Measurement Computing PCI-QUAD04 card. The torque output to the robot is provided by a Measurement Computing PCIM-DDA06/16

card with the same sample time. A computer monitor is placed in front of the subject to provide visual feedback about his/her motion trajectory during the execution of the task.

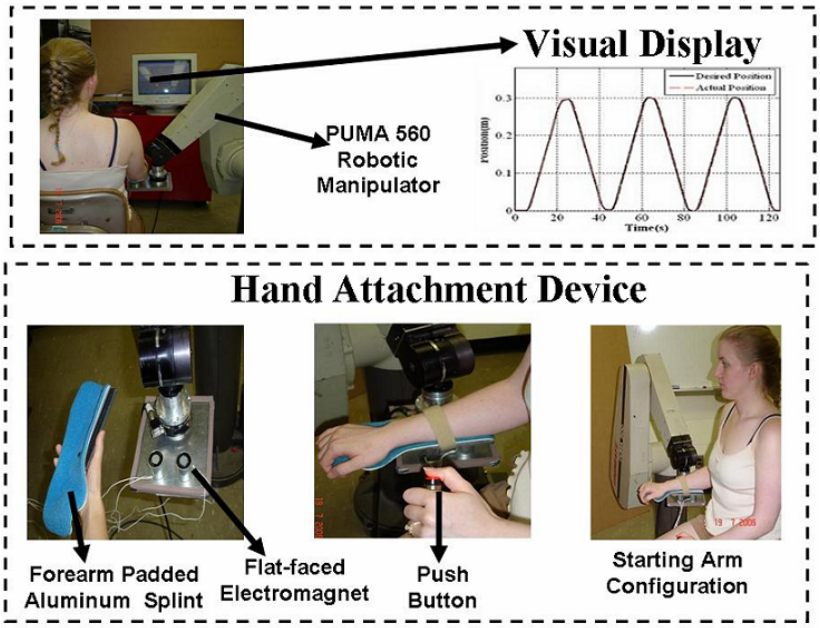


Fig. 1. Participant Arm attached to Robot

2.2. Hand Attachment Device

Since in this work we are primarily interested in effecting assistance to the upper arm, we design a hand attachment device where the participant's arm is strapped into a splint that restricts wrist and hand movement. The PUMA 560 is attached to that splint to provide assistance to the upper arm movement using the assistive controller (Fig. 1). Forearm padded aluminum splint (from MooreMedical), which ensures the participant's comfort, is used as a splint in this device. We further design a steel plate with proper grooves that hold two small flat-faced electromagnets (from Magnetool Inc.) that are screwed on it. This plate is also screwed with the force-torque sensor, which provides a rigid connection with the robot. We attach a light-weight steel plate under the splint, which is then attached to the electromagnets of the plate. These electromagnets are rated for continuous duty cycle (100% duty cycle), i.e., they can run continuously at normal room temperature. Pull ratings of these magnets are 40lb. We have used two electromagnets to have a larger pulling force to keep the splint attached to the hand attachment device. An automatic release (AU) rectifier controller (Magnetool Inc.) has been used to provide a quick, clean release of these electromagnets. A push button, which has been connected to the AU Rectifier Controller, is used to magnetize and demagnetize the

electromagnets when the participant wants to remove the hand attachment device from the robotic manipulator in a safe and quick manner.

2.3. Safety discussion about both the use of PUMA 560 Robotic Manipulator and Hand Attachment Device

Ensuring safety of the participant is a very important issue when designing a rehabilitation robotic system. Thus, in case of emergency situations, therapist can press emergency button. The patient and/or the therapist can quickly release the patient's arm from the PUMA 560 by using the quick-release hand attachment device (as described above) to deal with any physical safety related events. In order to release the participant's arm from the robot, the push button is used. When the push button is pressed electromagnets are demagnetized instantaneously and the participant is free to remove the splint from the robot. This push button can also be operated by a therapist. Additionally, we have covered the corner of the arm device with a foam self stick tape in order to avoid sharp surface.

3. Methodology

The objectives of the current work is to: i) design an upper arm movement rehabilitation task that requires cognitive processing as well as could contribute to a variety of functional daily living activities, and ii) design a controller to provide robotic assistance to help participants to perform the above movement rehabilitation task. In what follows we present the basic design of the task and the assistive controller.

3.1. Task Design

Let us first briefly review the task design of some well-known robotic rehabilitation systems. MIT Manus uses impedance controller to provide assistance to move the patient's arm to the target position in an active assisted mode, where patients can visually see their movement and target location (Krebs 2003, 2004). MIME provides assistance to move the participant's arm with a pre-programmed position trajectory using proportional-integral-derivative (PID) controller (Burgar 2000, Lum 2006). The participant is asked to maintain a specified off-axis force while they are trying to reach toward a goal position using ARM Guide (Kahn 2006). The GENTLE/s provides assistance to the patients to move to the target positions along with a predefined path using admittance control. The participant's movement trajectory is represented in the virtual environment in (Loureiro 2003). The therapy tasks

designed for the rehabilitation robotic devices require predominantly shoulder motion or elbow motion, or some of them require the combination of both shoulder and elbow motion.

We choose a reaching task that is commonly used for rehabilitation of upper extremity after stroke. In this task, the participants are asked to move their arms in the forward direction to reach a desired point in space and then bring it back to the starting position repeatedly within a specified time. In other words, they have to follow a desired position trajectory. The reaching task designed in here requires combination of the shoulder and elbow which could increase the active range of motion (AROM) in shoulder and elbow in preparation of later functional reaching activities in rehabilitation. The allowable motion is restricted only to the direction of the task. For example, if the task requires the participants to move their arms in the Y-direction, then they will not be able to move their arms in X or Z directions. However, they can move their arms in the Y-direction at a velocity that could be the same, higher or lower than the desired velocity. The idea here is to improve the ability of the participant's arm movement in one direction at a time by helping them to improve their speed of movement. Improving the speed of movement for such tasks is an important criterion to measure the success of a therapy. For example, in Constraint Induced Movement Therapy (CIMT) (Taub 1999) during the performance of the *wipe table* task, participants are required to complete as many back and forth motion as possible in a certain amount of time across the table and back between the two targets. The number of times of the completed movement in a certain amount of time is used as a metric to evaluate the participants' progress. If the participants can improve their speed of movement, the metric described above will capture this progress. In this work, we constrain the motion of the arm in the horizontal plane and in one direction (along the Y-axis). Although, in this work the motion of the arm is constrained in the horizontal plane in one direction (along the Y-axis), it could also be designed for other directions (e.g., X-axis) or combination of directions (e.g., XY-axes) based on task requirements (only shoulder or elbow motion or the combination of shoulder and elbow motion).

In order to include cognitive processing within this reaching task, we ask the participants to follow a visually presented desired motion trajectory that is likely to command their concentration. The participants receive visual feedback of both their actual position and the desired position trajectories on a computer screen, which is placed in front of them. They are asked to pay attention to tracking the desired position trajectory as accurately as possible, which keeps them focused on the task. The visual feedback is used not only to inform the participants of how closely they are tracking the desired motion but also as a motivational

factor to keep them focused on the task. The tip of the position trajectory that the participant is required to follow represents the velocity of the task trajectory.

The task presented here incorporates cognitive processing by asking the participants to follow the tip of the visually presented trajectory. The tip of the trajectory represents the current desired velocity. By asking the participant to follow the tip makes him/her focused on the task. This task is different from other tasks that have been used in the context of robotic rehabilitation in that here we are interested in improving the speed of motion in one direction at a time using visual feedback, which could be useful in a number of therapy tasks.

3.2. Controller Design

The controller designed in this work is responsible for providing robotic assistance to a participant to complete the movement tracking task in an accurate manner. The existing robotic rehabilitation systems operate in robot task-space to provide robotic assistance to the patients to follow a desired trajectory to complete a rehabilitation task (Krebs 2004, Lum 2006, Kahn 2006). Recently, a human-arm joint impedance controller is proposed, which operates in joint-space, to provide assistance to the subjects to follow desired joint angle trajectory (Culmer 2005) specified for each individual joint (e.g., elbow joint). It is still not clear, however, whether the assistance in the task-space or in the joint-space will likely to have the best results for rehabilitation purposes. In this work, we design a controller that is responsible for providing the robotic assistance to the subjects to complete a rehabilitation task in task-space. In this controller, an outer force feedback loop is designed around an inner position loop (Fig. 2). The tracking of the reference trajectory is guaranteed by the inner motion control (Sciavicco 1996). The desired force, which is given as a force reference to the controller, is computed by a planner. The proposed controller is similar to an impedance controller; however it allows specifying the reference time varying force directly. The equations of motion for the robot are given by:

$$\Gamma = M(q)\ddot{q} + Co(q)(q,\dot{q}) + Ce(q)\left|\dot{q}^2\right| + G(q) \qquad (1)$$

$$u - J^T(q)F = M(q)\ddot{q} + V(q,\dot{q}) + G(q)$$

where $M(q)$ represents the inertia matrix, $V(q,\dot{q})$ is the summation of the matrix of coriolis torques $Co(q)(q,\dot{q})$ and centrifugal torques $Ce(q)\left|\dot{q}^2\right|$, $G(q)$ is the vector of gravity torques. Γ is the generalized joint force torque which is calculated using $u - J^T(q)F$, where u is the input to the manipulator, $J(q)$ is the Jacobian matrix and F is the contact force

exerted by the manipulator. Using inverse dynamics control, manipulator dynamics are linearized and decoupled via a feedback. The dynamic equation of the robotic manipulator was given in (1). Control input u to the manipulator is designed as follows:

$$u = M(q)y + V(q,\dot{q}) + G(q) + J^T F \tag{2}$$

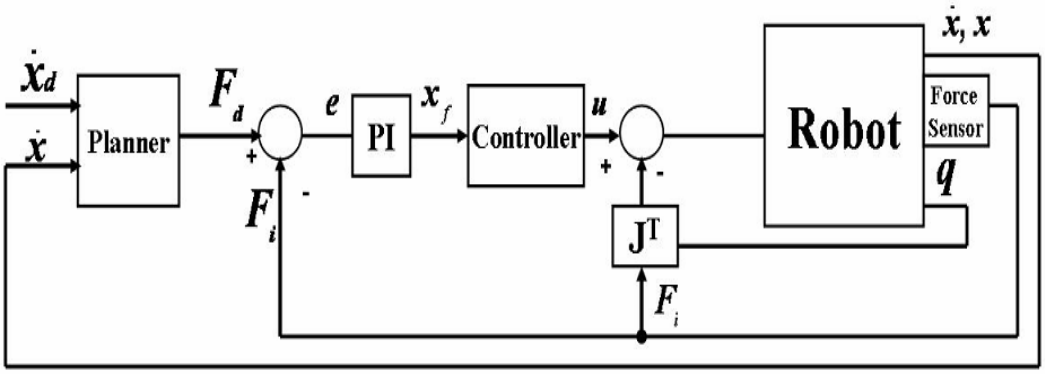


Fig. 2. Low-level Assistive Controller

which leads to the system of double integrators

$$\ddot{q} = y \tag{3}$$

In (3), y represents a new input. The new control input y is designed so as to allow tracking of the desired force F_d . To this purpose, the control law is selected as follows:

$$y = J(q)^{-1} M_d^{-1}(-K_d\dot{x} + K_p(x_f - x) - M_d\dot{J}(q,\dot{q})\dot{q}) \tag{4}$$

where x_f is a suitable reference to be related to force error. M_d (mass), K_d (damping) and K_p(stiffness) matrices specify the target impedance of the robot. x and $\dot{x}$ are the position and the velocity of the end-effector in the Cartesian coordinates, respectively. The relationship between the joint space and the Cartesian space acceleration is used to determine the position control equation:

$$\ddot{x} = J(q)\ddot{q} + \dot{J}(q,\dot{q})\dot{q} \quad and \quad \ddot{x} = J(q)y + \dot{J}(q,\dot{q})\dot{q} \tag{5}$$

By substituting (4) into (5), we obtain

$$\ddot{x} = J(q)(J(q)^{-1}M_d^{-1}(-K_d\dot{x} + K_p(x_f - x) - M_d\dot{J}(q,\dot{q})\dot{q})) + \dot{J}(q,\dot{q})\dot{q}$$

$$\ddot{x} = -M_d^{-1}K_d\dot{x} + M_d^{-1}K_p(x_f - x)$$

$$M_d\ddot{x} + K_d\dot{x} + K_p x = K_p x_f \tag{6}$$

Equation (6) shows the position control tracking of x with dynamics specified by the choices of K_d, K_p and M_d matrices. Impedance is attributed to a mechanical system characterized by these matrices that allows specifying the dynamic behavior. Let F_d be the desired force reference, which is computed using a PID velocity loop:

$$F_d = P_d(\dot{x}_d - \dot{x}) + I_d\int(\dot{x}_d - \dot{x})dt + D_d\frac{d(\dot{x}_d - \dot{x})}{dt} \tag{7}$$

where $\dot{x}_d, \dot{x}, P_d$, I_d and D_d are the desired velocity, actual velocity, the proportional, integral and derivative gains of the PID velocity loop, respectively. The relationship between x_f and the force error is expressed in (8) as:

$$x_f = P(F_d - F_i) + I\int(F_d - F_i)dt \tag{8}$$

where P and I are the proportional and integral gains, respectively, and F_i is the force applied by the human. Equations (6) and (8) are combined to obtain below equation:

$$M_d\ddot{x} + K_d\dot{x} + K_p x = K_p(P(F_d - F_i) + I\int(F_d - F_i)dt) \tag{9}$$

We can observe from (9) that the desired force response is achieved by controlling the position of the manipulator.

3.3. Decision of Robotic Assistance during Task Execution

During the tracking task, the activation of the low-level assistive controller to provide robotic assistance is decided based on the participant's actual velocity ($\dot{x}$). If the actual velocity lies within an acceptable band, then it is understood that the participant is able to track the trajectory without robotic assistance. The acceptable band consists of upper and lower bounds on velocity, which are defined as:

$$\dot{x}_{upper} = \dot{x}_d + \left(\dot{x}_d * \frac{percentage}{100}\right), \quad \dot{x}_{lower} = \dot{x}_d - \left(\dot{x}_d * \frac{percentage}{100}\right) \tag{10}$$

where *percentage* is the value used to increment and decrement the desired velocity to define the upper and the lower velocities for the selected $\dot{x}_d$. If the $\dot{x}$ is not between $\dot{x}_{upper}$ and $\dot{x}_{lower}$, then the low-level controller is activated to provide assistance to keep the participant's motion in the desired velocity range. However, note that any participant will require a finite amount of time to generate the desired motion. The controller should not be activated until it is determined that the participant is not able to generate the required motion by his/her own effort. Thus, initially a desired $\dot{x}_d$ is decided and it's upper ($\dot{x}_{upper}$) and

lower ($\dot{x}_{lower}$) bound is calculated using (10). In order to determine the velocity trajectories $\dot{x}_d(t)$, $\dot{x}_{upper}(t)$ and $\dot{x}_{lower}(t)$, we use a generator block using Matlab/Simulink Blockset. This block generates smooth velocity trajectories within a specified distance using a skew-sine function. As a result, we define an algorithm to determine the average velocity of the participant $\dot{x}_{ave}$ (as opposed to instantaneous velocity) and average value of the upper $\dot{x}_{upperave}$ and lower $\dot{x}_{lowerave}$ velocity bounds for a given period of time, which are used to decide if the robotic assistance is needed. $\dot{x}_{ave}$, $\dot{x}_{upperave}$ and $\dot{x}_{lowerave}$ are calculated using the equations:

$$\dot{x}_{ave} = \frac{1}{\left(\frac{tf-ti}{ts}\right)} \sum_{t=ti}^{tf} (\dot{x}(t)), \dot{x}_{lowerave} = \frac{1}{\left(\frac{tf-ti}{ts}\right)} \sum_{t=ti}^{tf} (\dot{x}_{lower}(t)), \dot{x}_{upperave} = \frac{1}{\left(\frac{tf-ti}{ts}\right)} \sum_{t=ti}^{tf} (\dot{x}_{upper}(t)) \quad (11)$$

where tf, ti and ts are the final time, the starting time and the sampling time, respectively. $\dot{x}(t)$ is the participant's actual velocity at time t. If $\dot{x}_{lowerave} < \dot{x}_{ave} < \dot{x}_{upperave}$ is satisfied, then the low-level controller is not activated and the participant continue tracking task without robotic assistance. If $\dot{x}_{lowerave} < \dot{x}_{ave} < \dot{x}_{upperave}$ is not satisfied then the controller is activated to provide robotic assistance to the participant to track the desired motion.

3.4. Switching Mechanism

Note that the controller will be switching in and out to provide robotic assistance. In order to ensure smooth switching, a switching mechanism that we have previously shown to guarantee bumpless switching for satisfactory force response (Mallapragada et al., 2006) is used in this work. This mechanism modifies the position reference, which is the input for the inner loop of the force controller, at the time of the switching in such a way that it is equal to the position reference at the time before switching occurred. The control action in (8) can be modified as below:

$$x_{fp}(t) = x(t) \quad and \quad x_{ff}(t) = Pe(t) + I(X_i(t) + X_{io})dt \quad (12)$$

Here $x_{fp}(t)$ is the position reference when the controller is not active, which is equal to the position of the human/robot $x(t)$. $x_{ff}(t)$ is the position reference determined using the P and I gains when the controller is active. $X_i(t)$ represents the integral action and X_{io} is the initial condition of the error integrator. $e(t)$ is defined as the $F_d - F_i$. If t_s is the time of

switching, then equation (12) can be used to find the position reference just before the time of switching:

$$x_{fp}(t_s^-) = x(t_s^-) \tag{13}$$

where $x(t_s^-)$ represents the position of the human/robot right before the switching occurred. The position reference just after the switching is given as:

$$x_{ff}(t_s^+) = Pe(t_s^+) + I(X_i(t_s^+) + X_{io})dt \tag{14}$$

The integral action associated with the controller is reset during the switching so that:

$$X_i(t_s^+) = 0 \tag{15}$$

The force error defined as $F_d - F_i$ is set to zero just after the time of the switching for a small period of time. Hence:

$$Pe(t_s^+) = 0 \tag{16}$$

After the time of the switching F_d which is calculated using (7), and F_i, which is recorded from the force sensor are provided to the controller. The initial condition X_{io} is defined as:

$$X_{io} = x(t_s^-)/I \tag{17}$$

Then, substituting (15)-(17) into (14) we can observe that

$$x_{ff}(t_s^+) = x_{fp}(t_s^-) \tag{18}$$

This relation ensures that the position reference is indeed continuous during switching which guarantees bumpless activation and deactivation of the controller.

4. Results

In this section we present the experiments and the results of the experiments with unimpaired participants.

4.1. Experiments

4.1.1. Experiment Procedure

Participants are seated in a height adjusted chair as shown in Fig. 1 (top left). The height of the PUMA 560 robotic manipulator has been adjusted for each participant to start the

tracking task in the same arm configuration. The starting arm configuration is selected as shoulder at neutral 0° position and elbow at 90° flexion position. The task requires moving the arm in forward flexion to approximately 60° in conjunction with elbow extension to approximately 0°. Participants are asked to place their forearm on the hand attachment device as shown in Fig. 1 (bottom left) when the starting arm configuration is fixed. The push button has been given to the participants that can be used during the task execution in case of emergency situations (Fig. 1- bottom middle). The participants receive visual feedback of their position on a computer monitor on top of the desired position trajectory (Fig. 1-top right). Participants were asked to execute the tracking task 50 times.

4.1.2. Description of the Experiments

We had conducted two experiments to evaluate the proposed assistive controller. In the first experiment, the participants were required to perform the tracking task without any external resistance applied to his/her upper arm. Participants were asked to track the position trajectory displayed on the computer screen. The participant's $\dot{x}_{ave}$ was calculated using (11) and if it was in between $\dot{x}_{upperave}$ and $\dot{x}_{lowerave}$ then the robot did not need to provide any assistance. However, friction and gravity compensation were always activated in order for the participant to move the robot along with his/her arm in an effortless way. If the $\dot{x}_{ave}$ was not between $\dot{x}_{upperave}$ and $\dot{x}_{lowerave}$, then assistive controller was activated to provide robotic assistance to complement the participant's effort to complete the task in a precise manner. During these two experiments, the number of trials and the number of times participant needed robotic assistance were recorded to observe the improvement of participant's movement ability.

In the second experiment, we asked the participant to perform the same task as in Experiment 1; however, in this case, the participant's arm movement ability was constrained with a resistive band (Thera-bands). This was done to simulate the movement of a stroke patient who may experience variable stiffness during the course of motion. In order to apply resistance to participant's upperarm, a mechanism was designed as shown in Fig. 3. Thera-bands are color-coded into many levels of resistance, thus different color resistive bands can be selected in order to simulate different stiffness of the stroke patient's arm. We selected the green (heavy) color resistive band for our experiment, because it provided sufficient resistance to the participant's movement while not inhibiting their ability to complete the

task. The mechanism has a rod which can slide right or left to change the position of the attachment and can be used for both right-handed and left-handed participants. The rod has holes on it to adjust the location of the resistive band on the upper arm that may vary among participants. The resistive band is connected to the participant's upper arm through a soft strapped attachment to prevent the participant's arm from the irritation that may be caused when the band is stretched. A seat-belt mechanism that connects the rod to the resistive band attachment can be used to release the rod from the resistive band quickly.

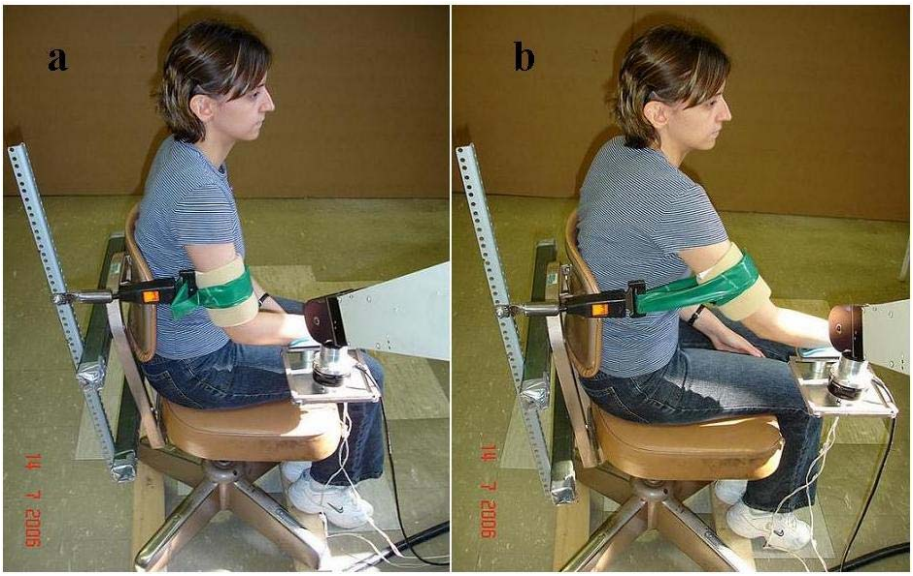


Fig. 3a. Initially No Resistance is applied to the Participant's Upper Arm, Fig. 3b. Resistance is applied to the Participant's Upper Arm in the Direction of Motion as the Task Begins

4.2. Results

Three female and one male participants within the age range of 25-30 years took part in the experiments that were described in above. All participants were right-handed. In these experiments (i.e., Experiments 1 and 2 as described in above), the participant tried to track the desired position trajectory by visually looking at the computer screen. Each participant performed the task 50 times for each experiment. $\dot{x}$ was selected as *0.02m/s*, which was chosen in consultation with a physical therapist who works with stroke patients at the Vanderbilt Stallworth Rehabilitation Hospital. The $\dot{x}_{upper}$ and $\dot{x}_{lower}$ were selected as *25%* more and less of $\dot{x}$, which were *0.025m/s* and *0.015m/s*, respectively. The range could be increased or decreased based on the participant's movement ability. Then, $\dot{x}_d(t)$, $\dot{x}_{upper}(t)$ and $\dot{x}_{lower}(t)$ velocity trajectories were generated using the reference block. The $\dot{x}_{ave}$,

$\dot{x}_{upperave}$ and $\dot{x}_{lowerave}$ were calculated using (11) at every *5* seconds. *5* seconds were sufficient to estimate the progress of the participant. If $\dot{x}_{lowerave} < \dot{x}_{ave} < \dot{x}_{upperave}$ was not satisfied then the controller was activated for the next *5* seconds to provide robotic assistance to the participant to track the desired motion within the desired velocity range.

In the first experiment (E1), each participant performed the tracking task without any external resistance applied to his/her upper arm. The idea was to assist the participants as and when they were out of the velocity band. It was noticed that the participants needed less assistance from the robot as they practiced more (Table 1). This result implies that the participants learned how to accomplish the task with practice.

Table 1. Number of Times Robot Assisted for E1

Trial Range / Number of Assistance for	1-10	11-20	21-30	31-40	41-50
P1	8	6	5	4	3
P2	14	13	13	12	11
P3	13	11	9	8	7
P4	12	12	11	10	9

Now we present the detailed analysis of the data for one participant (P1) as an example to demonstrate the effectiveness of the assistive controller. This data represented P1's 50^{th} trial. It could be observed from Fig. 4 that the participant's average velocity (dots), which was calculated every *5* seconds using (11), was out of range at *A*, *B* and *C* points. The controller was activated for the next *5* seconds to provide robotic assistance in order to take participant's velocity inside the velocity boundary, thus the controller was active between *A*-*A'*, *B*-*B*' and *C*-*C'* (Fig. 5). It could be seen that the participant's velocity was brought inside the desired range at *A'*, *B'* and *C'* points (Fig. 4), which verified that the assistive ability of the proposed controller.

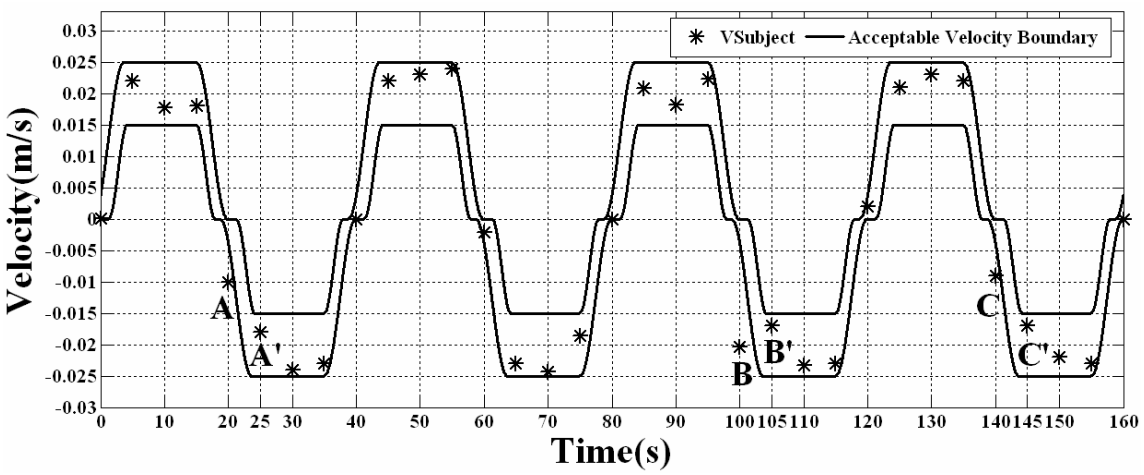


Fig. 4. Calculated Average Velocities for Experiment 1

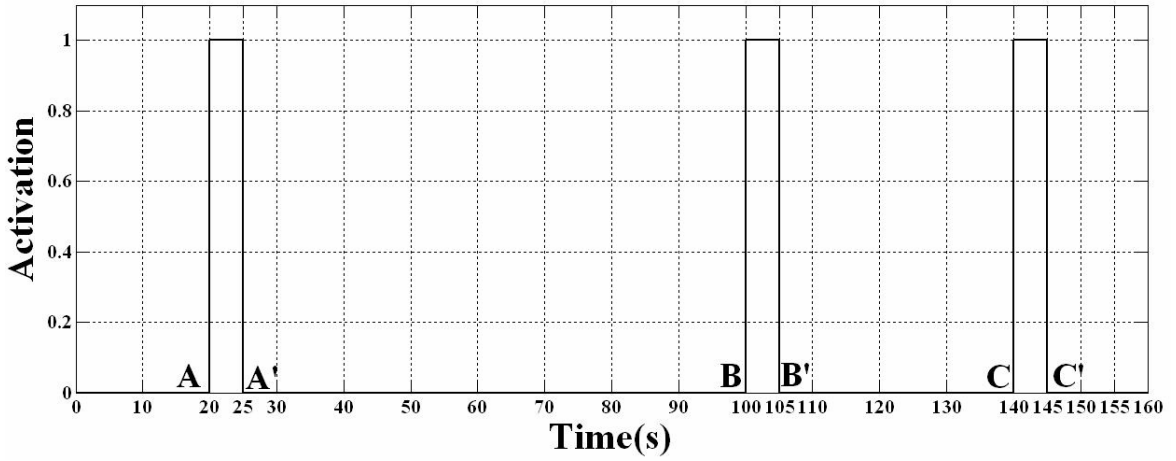


Fig. 5. Assistive Controller Triggering

We further analysed the amount of time taken by the assistive controller (t_s, in seconds) to take $\dot{x}$ into the desired velocity range. Here t_s is defined as the settling time, which is the time taken between the moment the controller was activated and the actual velocity reached the boundary of the desired velocity range. The mean and standard deviation of t_s for all participants' data for E1 are presented in Table 2.

Table 2. Settling Time for E1

Participant	Mean	Standard Deviation
P1	0.4723	0.1502
P2	0.5801	0.1937
P3	0.4929	0.1272
P4	0.545	0.232

Thus it can be observed from the above set of results that the proposed controller could assist as and when needed and the provided robotic assistance could quickly (i.e., in approximately *0.55* seconds) bring the participant's velocity in the desired range.

In the second experiment (E2), the participant's arm movement ability was constrained with a resistive band as shown in Fig. 3. The participants were asked to track the desired motion by visually looking at the screen as before. It was observed that the participants needed more robotic assistance when their motion was constrained. It could also be observed from Table 3, participants learned how to accomplish the task with practice. We present the mean and standard deviation of the settling time of the assistive controller in Table 4 for all participants' data when they performed E2.

Table 3. Number of Times Robot Assisted for E2

Trial Range / Number of Assistance for	1-10	11-20	21-30	31-40	41-50
P1	13	12	12	11	9
P2	16	16	14	13	13
P3	14	13	13	12	12
P4	15	15	14	13	13

The second experiment was conducted to observe the performance of the controller in an artificially constrained motion scenario, which might provide insight about applying the system to stroke patients whose movement could be naturally constrained. It can be observed that the controller was able to assist as and when needed and could bring the actual velocity of the participant's arm within the desired range in about *0.65* seconds.

Table 4. Settling Time for E2

Participant	Mean	Standard Deviation
P1	0.6317	0.232
P2	0.6274	0.2677
P3	0.6438	0.2674
P4	0.6985	0.248

5. Conclusion and Future Work

We have designed a movement tracking task where the participants not only make repetitive movement but also pay attention to the desired speed of motion from visual feedback. The task was designed in such a manner that it required cognitive processing. Including cognitive processing in the task design is an important criteria because it had been previously shown that the movement tracking task that requires cognitive processing achieved greater gains for brain reorganization of stroke patients than that of movement task that does not require cognitive processing (Carey 2005, 2006).

We have presented a controller to provide robotic assistance to participants to complete the movement tracking task. The assistive controller is evaluated with unimpaired participants. The results have demonstrated that the controller provided robotic assistance to participants as and when needed and quickly brought the participant's velocity in the desired range (i.e., in approximately *0.55* seconds for Experiment 1 and *0.65* seconds for Experiment 2).

We have used a switching mechanism to guarantee bumpless activation and deactivation of the controller.

We have shown that the participants needed less assistance from the robot as they practiced more, which implies that the participant's ability to complete the desired motion within a defined velocity range have been improved. Improving the velocity of patient's movement could be an important criterion to measure the success of a rehabilitation therapy.

We are aware that a PUMA 560 robot might not be ideal for rehabilitation applications. However the use of the hand attachment device, which has been described in Section 2, provided a quick release mechanism to protect the participant's arm from injuries. Note that the presented assistive controller is not specific to the proposed rehabilitation robotic system but can be integrated with any previously proposed rehabilitation robotic system.

An important direction for future development involves testing the usability of the proposed assistive robot controller with stroke patients. Functional magnetic resonance imaging (fMRI) procedure can be used to investigate whether the presented task that included cognitive processing result in long-term brain reorganization.

Acknowledgments

We gratefully acknowledge the help of Dr. Thomas E. Groomes who is the Medical Director of Spinal Cord and Traumatic Brain Injury Program, and therapist Sheila Davy of Vanderbilt University's Stallworth Rehabilitation Hospital for their feedback about

experimental setup and task design during this work. The work was supported by Vanderbilt University Discovery grant.

References

American Heart Association: Heart and Stroke Statistical Update (2006), Available from: http://www.Americanheart.org/statistics/stroke.htm.

Black, J.E.; Isaacs, K.R.; Anderson, B.J.; Alcantara, A.A. & Greenough, W. T. (1990). Learning causes synaptogenesis, whereas motor activity causes angiogenesis, in cerebellar cortex of adult rats. Proceedings of National Academic. Science, pp. 5568-5572, 87(15).

Burgar, C.G.; Lum, P.S.; Shor, P.C. & Van der Loos, H.F.M. (2000). Development of robots for rehabilitation therapy: The Palo Alto VA/Stanford experience. J. of Rehab. Research & Development, 37, 6, pp. 663-673.

Carey, J.R.; Bhatt, E. & Nagpal, A. (2005). Neuroplasticity Promoted by Task Complexity. Exercise and Sport Science Review, 33, pp. 24-31.

Carey J.; Durfee, W.; Bhatt, E. ; Nagpal, A.; Weinstein, S.;Anderson, K. & Lewis, S. (2006). Tracking vs. Movement Telerehabilitation Training to Change Hand Function and Brain Reorganization in Stroke. Submitted to Neurorehabilitation and Neural Repair.

Culmer, P. ; Jackson, A. ; Richardson, R. ; Bhakta, B. ; Levesley, M ; Cozens, A. (2005). An admittance control scheme for a robotic upper-limb stroke rehabilitation system. Int. Conf. On Engineering in Medicine and Biology Society, pp. 5081 - 5084.

Kahn, L.E.; Zygman, M.L; Rymer, W.Z. & Reinkensmeyer, D.J. (2006). Robot-assisted reaching exercise promotes arm movement recovery in chronic hemiparetic stroke: a randomized controlled pilot study. Journa of NeuroEngineering and Rehabilitation, .3, 12, pp. 1-13.

Kahn L.E; Lum P.S.; Rymer W.Z; Reinkensmeyer D.J. (2006). Robot-assisted movement training for the stroke-impaired arm: Does it matter what the robot does? J. of Rehab. Research & Development, 43(5):619-630.

Kleim, J. A.; Barbay, S. & Cooper, N.R. (2002). Motor learning-dependent synaptogenesis is localized to functionally reorganized motor cortex. Neurobio Lean Mem, 77, pp. 63-77.

Krebs, H.I.; Palazzolo, J.J.; Dipietro, L.; Ferraro, M.; Krol J.; Rannekleiv, K.; Volpe, B.T. & Hogan N. (2003). Rehabilitation Robotics: Performance-Based Progressive Robot-Assisted Therapy. Autonomous Robots, 15, 1, pp. 7-20.

Krebs, H. I.; Ferraro, M., Buerger, S.P., Newbery, M. J., Makiyama, A., Sandmann, M.; Lynch, D.; Volpe, B. T. & Hogan, N. (2004). Rehabilitation robotics: pilot trial of a spatial extension for MIT-Manus. Journal of NeuroEngineering and Rehabilitation. 1, 5, pp. 1-15.

Loureiro, R.; Amirabdollahian, F.; Topping, M.; Driessen, B. & Harwin, W. (2003). Upper limb mediated stroke therapy - GENTLE/s approach. Autonomous Robots. 15, pp. 35-51.

Lum, P.S.; Burgar, C.G.; Van der Loos, H.F.M.; Shor, P.C.; Majmundar M.; Yap R. (2006). MIME robotic device for upper-limb neurorehabilitation in subacute stroke subjects: A follow-up study . J. of Rehab. Research & Development, 43(5):631-642.

Mallapragada, V.; Erol, D. & Sarkar, N. (2006). A New Method for Force Control for Unknown Environments. Proceedings of the IEEE/RSJ Intl. Conf. On Intelligent Robots and Systems.

Matchar, D.B. & Duncan, P.W. (1994). Cost of stroke, Stroke Clin Updates, 5, pp. 9-12.

Nudo, R.; Milliken, G.; Jenkins, W. & Merzenich M. (1996). Use-dependent alterations of movement representations in primary motor cortex of adult squirrel monkeys. J. Neurosci, 16, 2, pp. 785-807.

Pascual-Leone, A.; Nguyen, K.T.; Kohen, A.D.; Brasil-Neto, J.; Cammarota, A. & Hallett M. (1995). Modulation of muscle responses evoked by transcranial magnetic stimulation during the acquisition of new fine motor skills. J. Neurophysiol. 74, pp. 1037-1045.

Plautz, E.J.; Milliken, G. W. & Nudo, R J. (2000). Effects of repetitive motor training on movement representations in adult squirrel monkeys: role of use versus learning. Neurobiol. Learn. Mem., 74, pp. 27-55.

PUMA 560 Related Sites on the Internet, Available from: www.ee.ualberta.ca/~jasmith/puma/pumasites.html.

Sciavicco, L. & Siciliano, B. (1996). Modeling and Control of Robot Manipulators, McGrawHill.

Taub, E.; Uswatte, G. & Pidikiti, R. (1999). Constraint-Induced Movement Therapy: A New family of techniques with broad application to physical rehabilitation – a clinical review. J. of Rehab. Research and Development. (36), pp. 237-251.

CHAPTER III: MANUSCRIPT 2

INTELLIGENT CONTROL FOR ROBOTIC REHABILITATION AFTER STROKE

Duygun Erol & Nilanjan Sarkar

(Submitted to Journal of Intelligent and Robotics Systems)

Abstract

The paper presents a new control approach to robot assisted rehabilitation for stroke patients. The control architecture is represented in terms of hybrid system model combining a high-level and a low-level assistive controller. The high-level controller is designed to monitor the progress and safety of the rehabilitation task. It also makes decisions on the modification of the task that might be needed for the therapy. We design a low-level assistive controller to provide robotic assistance for an upper arm rehabilitation task that works in coordination with the proposed high-level controller. Experimental results on unimpaired participants are presented to demonstrate the efficacy of both the high-level and low-level assistive controller.

Key words: robot-assisted rehabilitation, intelligent controller, movement tracking training

1. Introduction

Stroke is a highly prevalent condition especially among the elderly that results in high costs to the individual and society [1]. According to the American Heart Association, in the U.S., approximately 700,000 people suffer a first or recurrent stroke each year [2]. It is a leading cause of disability, commonly involving deficits of motor function. Recent clinical results have indicated that movement assisted therapy can have a significant beneficial impact on a large segment of the population affected by stroke or other motor deficit disorders. Experimental evidence suggests that intensive movement training of new motor tasks is required to induce long-term brain plasticity [3]. In the last few years, robot-assisted rehabilitation for rehabilitation of the stroke patients has been an active research area, which provide repetitive movement exercise and standardized delivery of therapy with the potential of enhancing quantification of the therapeutic process [4]-[10].

The first robotic assistive device used as a therapeutic tool, the MIT Manus [7][8] uses an impedance controller to provide assistance to move the patient's arm to the target position in an active assisted mode, where patients can visually see their movement and target location. Mirror Image Movement Enabler (MIME) [6] uses a PUMA 560 manipulator to provide assistance to move the subject's arm with a pre-programmed position trajectory using Proportional-Integral-Derivative (PID) controller. Assisted Rehabilitation and Measurement (ARM) Guide [5],[9], is another robotic system that is capable of generating both horizontal and vertical motion, which provides assistance or resistance to the patient's movement to complete the reaching task. The GENTLE/s is a commercially available haptic robot used to provide assistance to patients to move to the target positions along with a predefined path using admittance control. The subject's movement trajectory is represented in the virtual environment [10]. Studies with these robotic devices verified that robot-assisted rehabilitation results in improved performance of functional tasks.

The existing robotic rehabilitation systems primarily use low-level assistive controllers to assist the movement of patients' arms. For example, MIT Manus uses an impedance controller, MIME uses a PID controller and GENTLE/s uses an admittance controller for movement assistance. In some cases, the rehabilitation system keeps track of the status of the task (e.g., AutoCITE [11]). However, to our knowledge, none of these systems has a dedicated high-level controller that can comprehensively monitor the task, provide assessment of the progress, and alter the task parameters to impart effective therapy based on the patient's performance in an automated manner. Instead, in these existing robotic rehabilitation systems, a therapist administers the therapy where he/she monitors the progress of the tasks, patient's safety, and assess whether the task needs to be updated based on current condition of the therapy. As a result, it is likely to consume more time of the therapist, increase workload of the therapist, and consequently, increase the cost of treatment. In the current work, we present the design and development of a high-level controller that work in conjunction with the low-level assistive controllers such that it can determine the task updates dynamically based on patients' performance; and monitor the safety related events in an automated manner and generate an accommodating plan of action. This high-level controller is designed in such a manner that it can be extended without altering the low-level assistive controllers, by including more task status information and decision rules. It is expected, this augmentation of the low-level assistive controllers with a high-level controller will help the therapist to use his/her time more productively in higher-level decision making and possibly allowing him/her to supervise multiple patients simultaneously.

The primary focus of this paper is to present an intelligent controller, which is called a high-level controller, to monitor the progress of the task and to make decisions on the modification of the task that might be needed for the therapy. In this paper, we also design a low-level assistive controller that works in coordination with the high-level controller. Note that the presented high-level controller is not specific to a given low-level assistive controller but can be integrated with any previously proposed rehabilitation systems.

This paper is organized as follows. It first presents the overall control architecture in Section 2. Then the rehabilitation robotic system is presented in Section 3. The low-level assistive controller and the high-level controller of the overall control architecture have been described in Section 4 and Section 5, respectively. Results of the experiments are presented in Section 6 to demonstrate the efficacy of the high-level controller on unimpaired participants. Section 7 discusses potential contributions of this work and possible directions for future work.

2. Control Architecture

Let us first present the proposed control architecture of a robot-assisted rehabilitation system in the context of a rehabilitation task called reaching task. The reaching task designed in here requires combination of the shoulder and elbow movement, which could increase the active range of motion (AROM) in shoulder and elbow in preparation of later functional reaching activities in rehabilitation. In this task, the participants are asked to move their arms in the forward direction to reach a desired point in space and then bring it back to the starting position repeatedly within a specified time. In other words, they have to follow a desired motion trajectory. The participants receive visual feedback of both their actual motion and the desired motion trajectories on a computer screen, which is placed in front of them. They are asked to pay attention to tracking the desired motion trajectory as accurately as possible, which keeps them focused on the task. The visual feedback is used not only to inform the participants of how closely they are tracking the desired motion but also as a motivational factor to keep them focused on the task.

The stroke patients may not be able to track the desired motion trajectory because of their motor impairment. A low-level assistive controller could be used to provide robotic assistance to participants' arm movement as and when needed to help them to complete the reaching task. Note that various robot, human and general task related information, called events, could affect the reaching task. For example, if the robot joint motor develops any fault; or if the patient feels uncomfortable he/she might want to stop the task; or the patient is

more than capable of performing the current task and he/she needs more challenging task etc. These set of information may require some adjustments of the planning of the task. As a result, the low-level assistive controller also needs to be aware of these adjustments of the task to accomplish the therapy requirements.

In order to provide therapy that can accommodate the above requirements, a high-level controller could be used in conjunction with the low-level assistive controller that monitors the task and patient's safety and informs the low-level assistive controller about the task updates. The high-level controller in here plays the role of a human supervisor (therapist) who would otherwise monitor the task and assess whether the task needs to be updated. However, in general, the high-level controller and the low-level assistive controller cannot communicate directly because each may require different types of inputs and outputs. For example, a high-level controller may operate in the discrete domain whereas a low-level assistive controller may operate in the continuous domain. Thus an interface is required which can convert continuous-time signals to sequences of discrete values and vice versa. Hybrid system theory provides mathematical tools that can accommodate both continuous and discrete system in a unified manner. As a result, in this work, we take the advantage of using a hybrid system model to design our control architecture. A hybrid system model has three parts, a "Plant", a "Controller" (supervisor) and an Interface [12],[13]. In order to avoid confusion about terminology, we call the "Controller" in hybrid system model a high-level controller in this paper. The continuous part, identified as the "Plant" is the low-level assistive controller. Fig. 1 presents the proposed control architecture. There has been no work to our knowledge on designing such a hybrid system for rehabilitation purposes. However, in this paper, we argue that such a hybrid system framework could be useful in automating robotic rehabilitation and providing important aid to the therapist. Hybrid control framework has been effectively used in other fields, such as industrial robotics, medicine, and manufacturing [14]. Note that the presented control architecture is not specific to a reaching task but can be used for any other rehabilitation tasks.

In this architecture (Fig. 1), the state information from the robot and the human is monitored by the process-monitoring module through the interface to trigger the relevant events. Each event is represented as a plant symbol so that the high-level controller can recognize the event. Once the high-level controller receives the event through a plant symbol, the decision making module of the high-level controller generates sequences of control actions using its decision rules. The high-level controller is designed considering the need of the therapist and the patient and it can be easily modified and extended for new task

requirements. The decision of the high-level controller is sent to the low-level assistive controller through the interface using the control symbols. Interface converts the control symbols to the plant inputs which are used to update the task. The updated task is then executed by the low-level assistive controller. This cycle continues to complete the therapy.

The proposed control architecture is flexible and extendible in the sense that new events can be included and detected by simply monitoring the additional state information from the human and the robot, and accommodated by introducing new decision rules and new low-level assistive controllers.

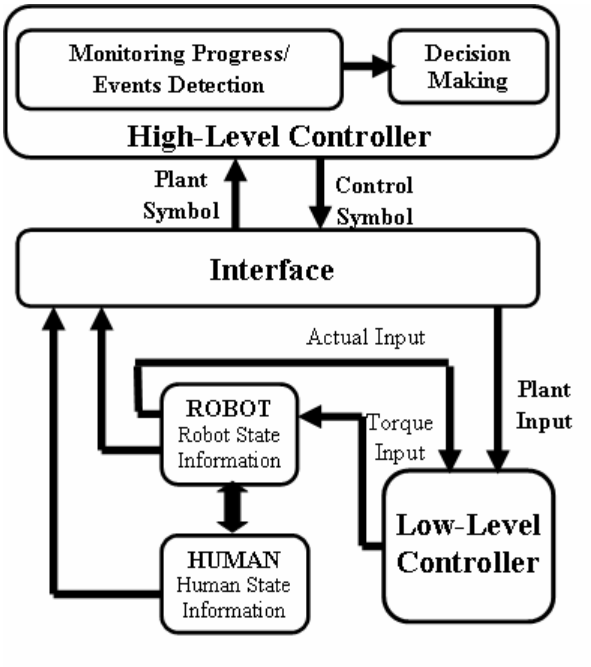


Figure 1 Control Architecture

3. Rehabilitation Robotic System

A PUMA 560 robotic manipulator is used as the main hardware platform in this work. The manipulator is augmented with a force-torque sensor and a hand attachment device (Fig. 2). The microcontroller board of the PUMA is replaced to develop an open architecture system to allow implementing the advanced controllers (e.g., low-level assistive and high-level controllers).

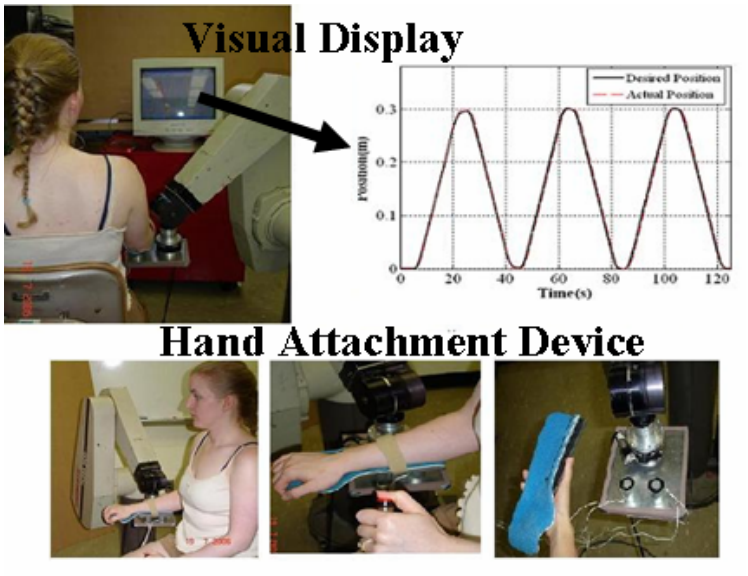


Figure 2 Participant Arm Attached to Robot

3.1. Hardware

The PUMA 560 is a 6 degrees-of-freedom (DOF) device consisting of six revolute axes. Each major axis (joints 1, 2 and 3) is equipped with electromagnetic brake, which is activated when power is removed from the motors, thereby locking the robot arm in a fixed position. The technical specifications of this robotic device can be found in [15]. In order to record the force and torque we have used ATI Gamma force/torque sensor. We have interfaced the robot with Matlab/Realtime Workshop to allow fast and easy system development. The force values recorded from the force/torque sensor are obtained using a National Instruments PCI-6031E data acquisition card with a sampling time of 0.001 seconds. The joint angles of the robot are measured using encoder readings with a sample time of 0.001 seconds from a Measurement Computing PCI-QUAD04 card. The torque output to the robot is given with a Measurement Computing PCIM-DDA06/16 card with the same sample time. A computer monitor is placed in front of the participant to provide visual feedback about his/her motion trajectory during the execution of the task.

3.2. Hand Attachment Device

Since in this work we are primarily interested in effecting assistance to the upper arm, we design a hand attachment device where the participant's arm is strapped into a splint that restricts wrist and hand movement. The PUMA 560 is attached to that splint to provide assistance to the upper arm movement using the assistive controller (Fig. 2). Forearm padded aluminum splint (from MooreMedical), which ensures the participant's comfort, is used as a

splint in this device. We further design a steel plate with proper grooves that hold two small flat-faced electromagnets (from Magnetool Inc.) that are screwed on it. This plate is also screwed with the force-torque sensor, which provides a rigid connection with the robot. We attach a light-weight steel plate under the splint, which is then attached to the electromagnets of the plate. These electromagnets are rated for continuous duty cycle (100% duty cycle), i.e., they can run continuously at normal room temperature. Pull ratings of these magnets are 40lb. We have used two electromagnets to have a larger pulling force to keep the splint attached to the hand attachment device. An automatic release (AU) rectifier controller (Magnetool Inc.) has been used to provide a quick release of these electromagnets. A push button, which has been connected to the AU Rectifier Controller, is used to magnetize and demagnetize the electromagnets when the participant wants to remove the hand attachment device from the robotic manipulator in a safe and quick manner.

3.3. Safety Discussion about both the Use of a PUMA 560 Robotic Manipulator and Hand Attachment Device

Ensuring safety of the participant is a very important issue when designing a rehabilitation robotic system. Thus, in case of emergency situations, therapist can press emergency button. The patient and/or the therapist can quickly release the patient's arm from the PUMA 560 by using the quick-release hand attachment device (as described above) to deal with any physical safety related events. In order to release the participant's arm from the robot, the push button is used. When the push button is pressed electromagnets are demagnetized instantaneously and the participant is free to remove the splint from the robot. This push button can also be operated by a therapist. Additionally, we have covered the corner of the arm device with a foam self stick tape in order to avoid sharp surface.

3.4. Kinematic and Dynamic Model

We first present the kinematic and dynamic model of the PUMA since these models are used for designing controllers used in this work. The forward kinematic equations of a PUMA 560 are available in the literature [16]-[18]. However, we have introduced a tool frame to include the location of the human arm through the hand attachment device. This frame has its origin located at the center of the splint with the same frame assignments as 4^{th} and 6^{th} joints of PUMA. The position and orientation of the human arm is calculated in the Cartesian coordinates using forward kinematics considering the tool frame. The forward kinematics equations used to calculate X, Y and Z are given in [19]. The Jacobian matrix

$J(q)$ used in the low-level assistive controller is found by taking the derivation of X, Y and Z with respect to first three joints of PUMA $q = [\, q_1\ q_2\ q_3\,]$.

$$J(q) = \left[\frac{\partial x}{\partial q_1} \middle| \frac{\partial x}{\partial q_2} \middle| \frac{\partial x}{\partial q_3} \right], x = \begin{bmatrix} X & Y & Z \end{bmatrix}^T \tag{1}$$

The orientation of the human arm is controlled by the last three joints of the PUMA. For the particular task considered in this work, we needed to keep the arm in a configuration such that it aligns with the direction of the motion. In order to achieve this objective we kept q_4 (4^{th} joint of PUMA) and q_5 (5^{th} joint of PUMA) at their initial values and the desired value of q_6 (i.e., q_{6d}) (6^{th} joint of PUMA) is calculated using (2), where $q_{6doffset}$ is the initial value of q_6. These three joint were separately controlled by a proportional-derivative (PD) controller because the low-level assistive controller was designed to provide positional assistance for the task.

$$q_{6d} = pi/2 - tan^{-1}(X, -Y) + q_{6doffset} \tag{2}$$

We have used a well known explicit model of the dynamics of the PUMA 560 arm [16], where the equations of motion for the robot are given in (3). $M(q)$ represents the inertia matrix, $V(q,\dot{q})$ is the summation of the matrix of coriolis torques $Co(q)(q,\dot{q})$ and centrifugal torques $Ce(q)\left|\dot{q}^2\right|$, and $G(q)$ is the vector of gravity torques. Γ is the generalized joint force torque which is calculated using $u - J^T F$, where u is the input to the manipulator, $J(q)$ is the Jacobian matrix and F is the contact force exerted by the manipulator. This model is used in the proposed low-level assistive controller (described later). The details of each term in (3) and the numerical values of the parameters of the model are given in [16].

$$\begin{aligned} \Gamma &= M(q)\ddot{q} + Co(q)(q,\dot{q}) + Ce(q)\left|\dot{q}^2\right| + G(q) \\ u - J^T(q)F &= M(q)\ddot{q} + V(q,\dot{q}) + G(q) \end{aligned} \tag{3}$$

4. Low-Level Assistive Controller

We design a low-level assistive controller that works in coordination with the high-level controller to perform the rehabilitation task. The high-level controller generates appropriate commands (which are presented in detail in Section 5) for the low-level assistive controller so that it can provide assistance to the participants to complete the rehabilitation task. In what follows we first present the basic design of the low-level assistive controller, and then

provide details of decision of the robotic assistance. Last, we present a switching mechanism that ensures the smoothness when the low-level assistive controller will be switching in and out to provide robotic assistance.

4.1. Controller Design

The controller designed in this work is responsible for providing robotic assistance to a participant to complete the movement tracking task in an accurate manner [19]. The existing robotic rehabilitation systems operate in robot task-space to provide robotic assistance to the patients to follow a desired trajectory to complete a rehabilitation task [4]-[10]. Recently, a human-arm joint impedance controller is proposed, which operates in joint-space, to provide assistance to the subjects to follow desired joint angle trajectory specified for each individual joint (e.g., elbow joint) [20]. It is still not clear, however, whether the assistance in the task-space or in the joint-space will likely to have the best results for rehabilitation purposes. In this work, we design a controller that is responsible for providing the robotic assistance to subjects to complete a rehabilitation task in task-space. In this controller, an outer force feedback loop is designed around an inner position loop. The outer force control is in charge of computing a suitable reference end-effector trajectory, which ensures a compliant behavior of the manipulator when the end effector interacts with the participant. The tracking of the reference trajectory is guaranteed by the inner motion control loop [21] (Fig. 3). The desired force, which is given as a force reference to the force controller, is computed by a planner. In the planner, velocity error is transformed into a force reference using an outer PID velocity loop.

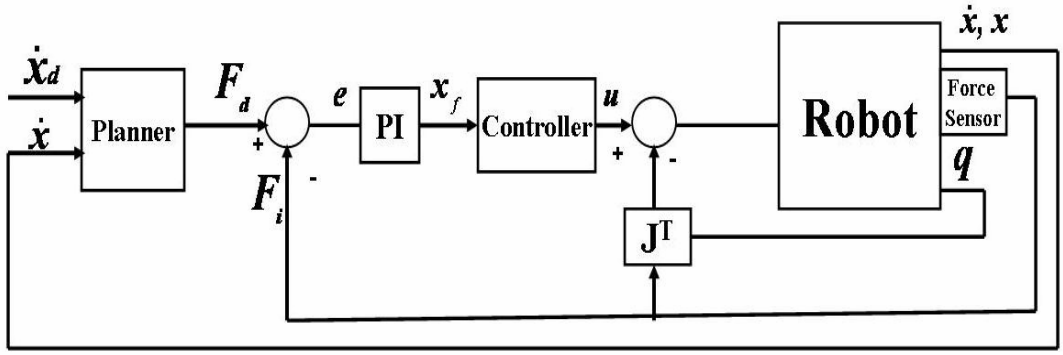


Figure 3 Low-Level Assistive Controller

Using inverse dynamics control, manipulator dynamics are linearized and decoupled via a feedback. The dynamic equation of the robotic manipulator was given in (3). Control input u

to the manipulator is shown in (4). Equation (4) leads to the system of double integrator, $\ddot{q} = y$, where y represents a new input.

$$u = M(q)y + V(q,\dot{q}) + G(q) + J^T F \tag{4}$$

The new control input y is designed so as to allow tracking of the desired force F_d. To this purpose, the control law is selected as given in (5). x_f is a suitable reference to be related to force error. M_d (mass), K_d (damping), and K_p (stiffness) matrices specify the target impedance of the robot. x and $\dot{x}$ are the position and velocity of the end-effector in the Cartesian coordinates, respectively.

$$y = J(q)^{-1} M_d^{-1}(-K_d \dot{x} + K_p(x_f - x) - M_d \dot{J}(q,\dot{q})\dot{q}) \tag{5}$$

The relationship between the joint and the Cartesian space acceleration is used to determine position control equation.

$$\ddot{x} = J(q)y + \dot{J}(q,\dot{q})\dot{q} \tag{6}$$

By substituting (5) into (6), we obtain (7). Equation (7) shows the position control tracking of x with dynamics specified by the choices of, M_d, K_d and K_p matrices. Impedance is attributed to a mechanical system characterized by these matrices that allows specifying the dynamic behavior.

$$M_d \ddot{x} + K_d \dot{x} + K_p x = K_p x_f \tag{7}$$

F_d, which is the desired force reference, is computed using a PID velocity loop, where $\dot{x}_d, \dot{x}, P_d, I_d$ and D_d are the desired velocity, actual velocity, the proportional, integral and derivative gains of the PID velocity loop, respectively.

$$F_d = P_d(\dot{x}_d - \dot{x}) + I_d \int(\dot{x}_d - \dot{x})dt + D_d \frac{d(\dot{x}_d - \dot{x})}{dt} \tag{8}$$

The relationship between x_f and the $F_d - F_i$ is expressed in (9), where P and I are the proportional and integral gains, respectively, and F_i is the interaction force between robot and the human.

$$x_f = P(F_d - F_i) + I\int(F_d - F_i)dt \tag{9}$$

Equations (7) and (9) are combined to obtain:

$$M_d \ddot{x} + K_d \dot{x} + K_p x = K_p(P(F_d - F_i) + I\int(F_d - F_i)dt) \tag{10}$$

We can observe from (10) that the desired force response is achieved by controlling the position of the manipulator.

4.2. Decision of Robotic Assistance during Task Execution

During the tracking task, the activation of low-level assistive controller to provide robotic assistance is decided based on the participant's actual velocity, $\dot{x}$. The participants will require a finite amount of time to generate the desired motion. The robotic assistance should not be activated until it is determined that the participant is not able to generate the required motion by his/her own effort. Thus, initially the desired $\dot{x}_d$ is decided and then it's upper $(\dot{x}_u)$ and lower $(\dot{x}_l)$ bounds are calculated using (11). The average velocity of the participant $\dot{x}_{ave}$ (as opposed to instantaneous velocity) and average value of the upper $(\dot{x}_{uave})$ and lower $(\dot{x}_{lave})$ velocity bounds for a given period of time are calculated using (11) to decide if the robotic assistance is needed. The $percentage$ is the value used to define the upper and lower velocities for the selected $\dot{x}_d$. tf, ti and ts are the final time, starting time and sampling time, respectively.

$$\dot{x}_u = \dot{x}_d + \left(\dot{x}_d * \frac{percentage}{100} \right),\ \dot{x}_l = \dot{x}_d - \left(\dot{x}_d * \frac{percentage}{100} \right)$$
$$\dot{x}_{ave} = \frac{1}{\left(\frac{tf - ti}{ts} \right)} \sum_{t=ti}^{tf} \left(\dot{x}(t) \right),\ \dot{x}_{uave} = \frac{1}{\left(\frac{tf - ti}{ts} \right)} \sum_{t=ti}^{tf} \left(\dot{x}_u(t) \right),\ \dot{x}_{lave} = \frac{1}{\left(\frac{tf - ti}{ts} \right)} \sum_{t=ti}^{tf} \left(\dot{x}_l(t) \right) \quad (11)$$

If $\dot{x}_{lave} < \dot{x}_{ave} < \dot{x}_{uave}$ is satisfied, then the low-level assistive controller is not activated and the participant continue tracking task without robotic assistance. If $\dot{x}_{lave} < \dot{x}_{ave} < \dot{x}_{uave}$ is not satisfied then the low-level assistive controller is activated to provide assistance to the participant to track the desired motion.

4.3. Switching Mechanism

Note that the low-level assistive controller will be switching in and out to provide robotic assistance. In order to ensure smooth switching, a switching mechanism that we have previously shown to guarantee bumpless switching for a satisfactory force response [22] is used in this work. This mechanism modifies the position reference, which is the input for the inner loop of the low-level assistive controller, at the time of the switching in such a way that it is equal to the position reference at the time before switching occurred. The control action

in (9) can be modified as (12), where the $x_{fp}(t)$ is the position reference determined when the low-level assistive controller is not active, which is equal to the position of the tool frame $x(t)$. $x_{ff}(t)$ is the position reference determined using the P and I gains when the low-level assistive controller is active. $X_i(t)$ represents the integral action and X_{io} is the initial condition of the error integrator. $e(t)$ is defined as the $F_d - F_i$.

$$\begin{aligned} x_{fp}(t) &= x(t) \\ x_{ff}(t) &= Pe(t) + I(X_i(t) + X_{io})dt \end{aligned} \tag{12}$$

If t_s is the time of switching, the position reference can be found using (13). $x(t_s^-)$ represents the position of the tool frame just before the switching occurred.

$$\begin{aligned} x_{fp}(t_s^-) &= x(t_s^-) \\ x_{ff}(t_s^+) &= Pe(t_s^+) + I(X_i(t_s^+) + X_{io})dt \end{aligned} \tag{13}$$

The integral action associated with the low-level assistive controller is reset during the switching $X_i(t_s^+) = 0$. The $F_d - F_i$ is set to zero just after the time of the switching for a small period of time $Pe(t_s^+) = 0$. The initial condition is set as $X_{io} = x(t_s^-)/I$. When these definitions are substituted in (13) we get (14). (14) ensures that the position reference is indeed continuous during switching, which guarantees bumpless activation and deactivation of the low-level assistive controller.

$$x_{ff}(t_s^+) = x_{fp}(t_s^-) \tag{14}$$

5. The High-Level Controller

The high-level controller monitors the progress of the task, the status of the plant, and makes decision on the modification of the task that might be needed for the therapy. The decision of the high-level controller is executed by the low-level assistive controller to accomplish the task requirements. In this section, we first present the theory of the high-level controller, followed by the design rationale and details of the high-level controller.

5.1. Theory

The high-level controller is a discrete-event system (DES) deterministic finite automaton, which is specified by $D = (\tilde{P}, \tilde{X}, \tilde{R}, \psi, \lambda)$ [12],[13]. Here $\tilde{P}$ is the set of discrete states. Each

event is represented as a plant symbol, where $\tilde{X}$ is the set of such symbols, for all discrete states. The next discrete state is activated based on the current discrete state and the associated plant symbol using the following transition function: $\psi : \tilde{P} \times \tilde{X} \to \tilde{P}$. In order to notify the low-level assistive controller the next course of action in the new discrete state, the controller generates a set of symbols, called control symbols, denoted by $\tilde{R}$, using an output function: $\lambda : \tilde{P} \to \tilde{R}$. The action of the high-level controller is described in (15), where $\tilde{p}_i, \tilde{p}_j \in \tilde{P}$, $\tilde{x}_k \in \tilde{X}$ and $\tilde{r}_c \in \tilde{R}$. i and j represent the index of discrete states. k and c represent the index of plant symbols and control symbols, respectively. n is the time index that specifies the order of the symbols in the sequence.

$$\begin{aligned} \tilde{p}_j[n] &= \psi(\tilde{p}_i[n-1], \tilde{x}_k[n]) \\ \tilde{r}_c[n] &= \lambda(\tilde{p}_j[n]) \end{aligned} \tag{15}$$

5.2. Design Rationale for the High-Level Controller

Let us explain the role of each element of the automaton $D = (\tilde{P}, \tilde{X}, \tilde{R}, \psi, \lambda)$ in the context of rehabilitation tasks. $\tilde{P}$ is the set of discrete states. A rehabilitation therapy may consist of several actions and each discrete state may capture one of these actions. The action that takes place in each discrete state could be used to update the rehabilitation task. For example, if improving the speed of motion is the objective, then each category of speed (e.g., slow, medium, fast etc.) could be chosen as discrete states. When new actions are required for a rehabilitation task, new states can easily be included in the set of the states, $\tilde{P}$. Once the set $\tilde{P}$ is chosen, the next design parameters are what are called “events” that could affect the rehabilitation task. Events are various robot, human and general task related information that provide the current status of the task. The set of events are not unique and are decided considering the need of the therapy, and the capabilities of the rehabilitation robotic systems. Generally the available sensory information from the robotic systems and the input from the therapist and the participant provide the core of the set of the events. When these events occur it may require some adjustments of the planning of the rehabilitation task. As discussed earlier, this sensory information may not be directly interpreted by the high-level controller. As a result, each event is represented as a plant symbol so that the high-level controller can recognize the events. $\tilde{X}$ is the set of the plant symbols, which is designed based on the set of events. The transition function $\psi : \tilde{P} \times \tilde{X} \to \tilde{P}$ uses the current state and the plant symbol to determine the next action that is required to update the rehabilitation task. For example, when

the participant is performing the rehabilitation task and an event that requires the task to be stopped occurs, then the transition function is used to transit from one active state, which executes the task as required, to another one, which stops the task execution, based on the event. The high-level controller generates a control symbol, which is unique for each state, using the output function λ. $\tilde{R}$ is the set of the control symbols. The output of the control symbols are plant inputs which is in charge of the modification of the rehabilitation task. The control symbols and its outputs are decided based on the task requirements and the abilities of the low-level assistive controller. For example, if the objective of the rehabilitation task is to increase the participant's range of motion, then the control symbol generates plant inputs to the low-level assistive controller to change the desired goal position of the task in order to make the task more/less challenging for the participant. It is clear from the above discussion that the design of the various elements of the automaton $D=(\tilde{P},\tilde{X},\tilde{R},\psi,\lambda)$ is not unique and is dependent on the task at hand, and sensory information available from the robotic system. In what follows we present the design of these elements with regard to the objective of the rehabilitation task we present in this paper.

The design of the elements of $D=(\tilde{P},\tilde{X},\tilde{R},\psi,\lambda)$ for the reaching task that has been described in Section 2 is motivated by the specific objective of the task. In here, the objective of the reaching task is to improve the participant's speed of movement while considering the current movement ability of the participant and the safety of the task. The participant is required to complete the movement in a certain amount of time, which represents the velocity of the task trajectory. The desired velocity trajectory could be updated to improve the participant's speed of movement and to ensure the safety of the participant. Thus the discrete states could be the level of speed at which the therapy is imparted to the participant. In order to decide the set of events, all sensory information that the current rehabilitation robotic systems can generate is analyzed. The rehabilitation robotic system used in this work has a force sensor to record the applied force by the participant, a PCI card to record the robot joint angles, and pause, stop and restart buttons for task execution. A counter is also used to record the number of times a participant needed robotic assistance to determine the improvement of participant's movement ability. This set of information is used to define several events in our work. Once the discrete states and the events are determined, the necessary plant and control symbols are designed based on the structure of the high-level and low-level assistive controllers, and the objectives of the task (e.g., when should discrete states be changed, how

to increase or decrease speed etc.). The design details of the high-level controller for the reaching task are given in the next section.

5.3. Design Details of the High-Level Controller

We initially define the following plant states $\tilde{p}$: stay, difficult, easy, stop and pause. Stay $(\tilde{p}_1)$ implies the participant needs to continue the task at the same difficulty level by keeping the desired velocity same. Difficult $(\tilde{p}_2)$ means the participant has improved his/her task performance and task need to be more challenging by increasing the desired velocity. Similarly, easy $(\tilde{p}_3)$ implies changing the task parameters to make the task easier by decreasing the desired velocity. Stop $(\tilde{p}_4)$ and pause $(\tilde{p}_5)$ are defined in their usual ways. New plant states can easily be included in the design of the high-level controller when new control actions are needed to modify the task parameters.

The state information from the robot and the human is decided to define the events. The state information from the robot and the human can be a continuous signal or a discrete value. Let S_{Rn} and S_{Hn} represent the sets of robot and human state information, respectively. In this research, the continuous signals that are detected from the robot are: i) robot's joint angles (S_{R1}), ii) the force reference calculated using (8) (S_{R2}), iii) the participant's velocity, which is measured from the tool frame velocity (S_{R3}). The discrete value detected from the robot is the participant's progress during the tracking task (S_{R4}). In order to find S_{R4}, the number of times participant needed robotic assistance at 10th trial (n_{10}) and at 50th trial (n_{50}) were recorded. Decision logic is defined to determine the value of S_{R4} using (16). Δp is the percentage change of velocity that can be defined by the therapist and it can be person-specific.

$$\begin{aligned} &if \quad n_{50} < \left(n_{10} - \left(n_{10} * \frac{\Delta p}{100}\right)\right) \quad then \quad \{S_{R4}=1\} \\ &elseif \quad n_{50} > \left(n_{10} + \left(n_{10} * \frac{\Delta p}{100}\right)\right) \quad then \quad \{S_{R4}=-1\} \\ &else \quad \{S_{R4}=0\} \end{aligned} \tag{16}$$

Robot and human state information is monitored to trigger relevant events to modify the task. When these events are triggered, the interface provides the necessary plant symbol $(\tilde{x})$ to the high-level controller. Currently we have defined nine events for the proposed high-level controller. However, the number of events can be easily extended. Five of these (*E1, E2, E3, E4* and *E5*) are robot generated, and three of these (*E6, E7* and *E8)* are human generated events. The other event, which is a secondary event, is called *SE1*. This is used to detect the previous state when the participant wants to continue with the task after he/she

stops. The high-level controller needs to know which state was active before the pause or stop button was pressed in order to provide the same task parameters to the participant when he/she resumes the task. For example, when the participant presses pause button, a value is assigned to *SE1*. This value is retrieved when the participant resumes the task so that he or she can continue the therapy with the same task requirements. Events are reset at the beginning of task execution. Additionally, the triggered event is reset when a new event occurs. When the participant requires less, more or same level of robotic assistance to track the desired trajectory, *E1*, *E2* and *E3* is triggered, respectively. *E4* occurs when the robot's joint angles are out of range. If the force reference (calculated by (8)) provided to the low-level assistive controller to assist the participant and the participant's velocity $(\tilde{x})$ are above predefined threshold values, then *E5* and *E6* are triggered, respectively. *E7* occurs when the participant presses the pause or the stop button. In order to continue with the task, the participant resets the pause button and *E8* event is triggered. Plant symbols $(\tilde{x})$ are designed based on the events as shown in Table 1. The $joint\_limits$ are known from the robot's specifications. $F_{dthreshold}$ and $\dot{x}_{threshold}$ are determined by the therapist at the beginning of the task execution. Note that if any of *E4*, *E5*, *E6*, and *E7* or their combinations occurs then the state *stop* $(\tilde{p}_4)$ is activated. Thus we assign the same plant symbol, $\tilde{x}_4$ for these events**.**

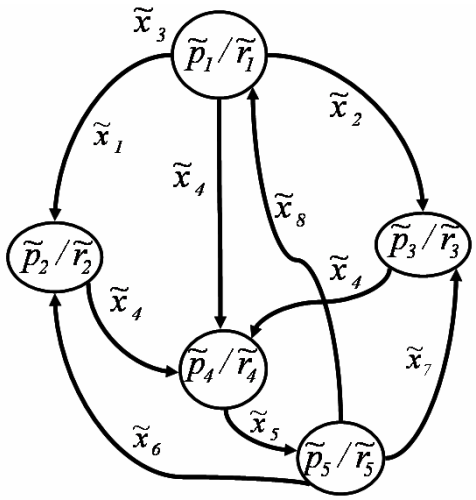
Table 1 Plant Symbols for the High-Level Controller

Signals from Human and Robot	Event Triggered	Plant Symbol
$S_{R4}=1$	$E1=1$	$\tilde{x}_1$
$S_{R4}=-1$	$E2=1$	$\tilde{x}_2$
$S_{R4}=0$	$E3=1$	$\tilde{x}_3$
$S_{R1}>joint\_limits$ or $S_{R2}> F_{dthreshold}$ or $S_{R3}> \dot{x}_{threshold}$ or $S_{H1}=1$	$E4=1$ $E5=1$ $E6=1$ $E7=1$	$\tilde{x}_4$
$S_{H2}=1$	$E8=1$	$\tilde{x}_5$

The secondary event, *SE1*, is defined as follows: if the state is *difficult* and *E7*=1, then *SE1*=1. We assign a corresponding plant symbol $\tilde{x}_6$. Similarly, if the state is *easy* and *E7*=1,

then *SE1*=2, and the plant symbol $\tilde{x}_7$ is assigned. If the state is *stay* and *E7*=1, then *SE1*=3. We assign a corresponding plant symbol $\tilde{x}_8$. *SE1* releases state information when *E7*=0 and *E8*=1.

When any of these events is triggered, the high-level controller decides the next plan of action to modify the task. When an event is triggered, the corresponding plant symbol $(\tilde{x})$ is generated by the interface. The current state $(\tilde{p})$ and the plant symbol $(\tilde{x})$ are used by the high-level controller to determine the next state. Then the high-level controller generates the corresponding control symbol $(\tilde{r})$ for this new state and provides it to the interface. The control mechanism of the proposed high-level controller is shown in Fig. 4 (left). In this figure, $\tilde{r}_c$ s are corresponding control symbols for each plant symbol $\tilde{x}_k$, where *c=1,2,…5* and *k=1,2,…8*. Any event that generates corresponding plant symbols $\tilde{x}_k$ along with the current state information $\tilde{p}_i$ determines the next $\tilde{p}_j$ and as a result, $\tilde{r}_c$, where *i=1,2,…5* and *j=1,2,…5*. In our application only one state is active at any given time, and therefore we uniquely assign a control symbol $\tilde{r}_i$ for each discrete state $\tilde{p}_i$. Since the low-level assistive controller cannot interpret the control symbols, the interface converts them to the appropriate values for α and β for (17) to execute the task. The available control symbols $\tilde{r}_i$ and their corresponding α and β values for the plant input are defined in a table in Fig. 4 (right).



Control Symbol	Constant Value
$\tilde{r}_1$	α=0,β=1
$\tilde{r}_2$	α=1,β=1
$\tilde{r}_3$	α=-1,β=1
$\tilde{r}_4$	β=0
$\tilde{r}_5$	β=0

Figure 4 Control Mechanism for the High-Level Controller

The plant equation which determines the desired velocity for the low-level assistive controller is defined in (17), where $delta$ is selected as a constant value to increase and

decrease the $\dot{x}_d$, which makes the task more or less challenging. $\dot{x}_{dm}$ is the new desired velocity value used to determine the new $\dot{x}_u$ and $\dot{x}_l$.

$$\dot{x}_{dm} = \beta(\dot{x}_d + (\alpha * delta)) \tag{17}$$

Then $\dot{x}_{ave}$, $\dot{x}_{uave}$ and $\dot{x}_{lave}$ are calculated using (11). If $\dot{x}_{lave} < \dot{x}_{ave} < \dot{x}_{uave}$ is not satisfied then the low-level assistive controller is activated to provide assistance to complement participant's effort to complete the task in a precise manner. The Matlab/Simulink/Stateflow software is used to implement the proposed high-level controller [23] (Fig. 5).

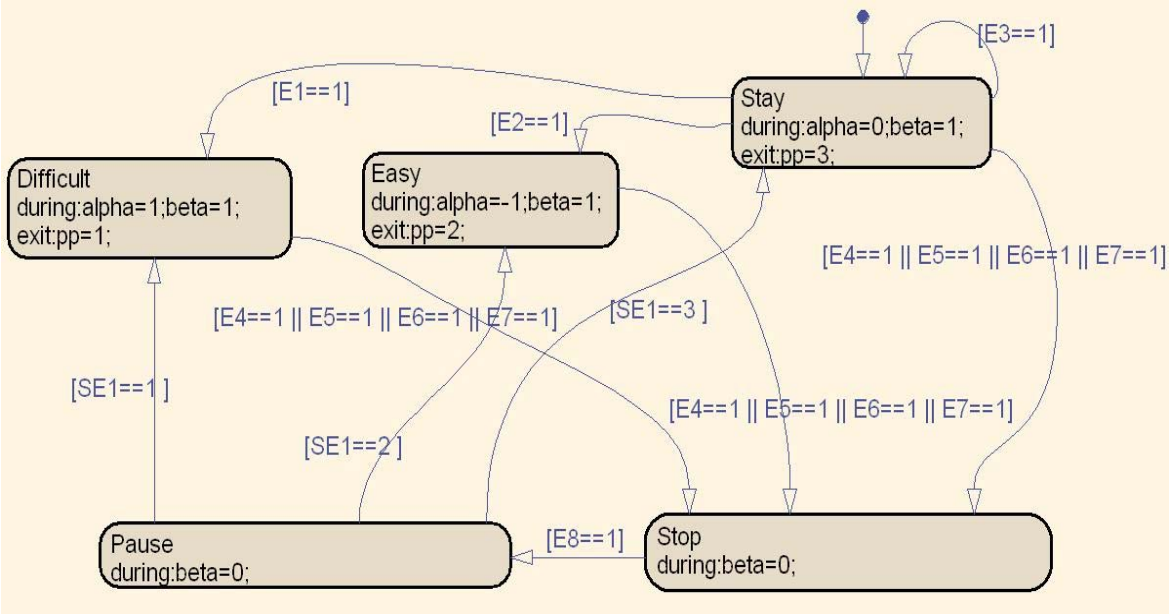


Figure 5 Stateflow Model for the High-Level Controller

6. Results

In this section we present the experimental procedure and the results of the experiments with unimpaired participants.

6.1. Experiment Procedure

Participants are seated in a height adjusted chair as shown in Fig. 2 (top left). The height of the PUMA 560 robotic manipulator has been adjusted for each participant to start the tracking task in the same arm configuration. The starting arm configuration is selected as shoulder at neutral $0°$ position and elbow at $90°$ flexion position. The task requires moving the arm in forward flexion to approximately $60°$ in conjunction with elbow extension to

approximately *0°*. Participants are asked to place their forearm on the hand attachment device as shown in Fig. 2 (bottom left) when the starting arm configuration is fixed. The push button has been given to the participants that can be used during the task execution in case of emergency situations (Fig. 2 - bottom middle). The participants receive visual feedback of their position on a computer monitor on top of the desired position trajectory (Fig. 2-top right). Participants are asked to execute the tracking task 50 times.

6.2. Results

6.2.1. Low-Level Assistive Controller Evaluation

Three female and one male participants within the age range of 25-30 years, right-handed participants took part in the experiment. The participants tried to track the desired position trajectory by visually looking at the computer screen. $\dot{x}$ was selected as *0.02m/s*, which was chosen in consultation with a physical therapist who works with stroke patients at the Vanderbilt Stallworth Rehabilitation Hospital. The $\dot{x}_u$ and $\dot{x}_l$ were selected as *25%* more and less of $\dot{x}$, which were *0.025m/s* and *0.015m/s*, respectively. The range could be increased or decreased based on the participant's movement ability. Then $\dot{x}_{ave}$, $\dot{x}_{uave}$ and $\dot{x}_{lave}$ were calculated using (11) at every *5* seconds. *5* seconds were sufficient to estimate the progress of the participant. Thus, the condition $\dot{x}_{lave} < \dot{x}_{ave} < \dot{x}_{uave}$ was checked to decide the activation of the low-level assistive controller. The idea of the low-level assistive controller was to assist the participants as and when they were out of the velocity band.

Participants were asked to execute the tracking task 50 times. The number of trials and the number of times participant needed robotic assistance were recorded. Friction and gravity compensation were always activated in order for the participant to move the robot along with his/her arm in an effortless way. We had noticed that the participants needed less assistance from the robot as they practiced more (Table 2), which implies that the participants learned how to accomplish the task with practice.

Table 2 Number of Times Robot Assisted

Trial Range / Number of Assistance for	1-10	11-20	21-30	31-40	41-50
P1	8	6	5	4	3
P2	14	13	13	12	11
P3	13	11	9	8	7
P4	12	12	11	10	9

We represent P1's 50th trial data as an example to show the task performance. It could be observed from Fig. 6 that the P1's average velocity (stars), which was calculated every 5 seconds using (11), was out of range at *A, B* and *C* points (Fig. 6). The low-level assistive controller was activated for the next 5 seconds between *A-A', B-B'* and *C-C'*. It could be seen that the P1's velocity was brought inside the desired range at *A'*, *B*' and *C'* points (Fig. 6), which verified the assistive ability of the low-level assistive controller.

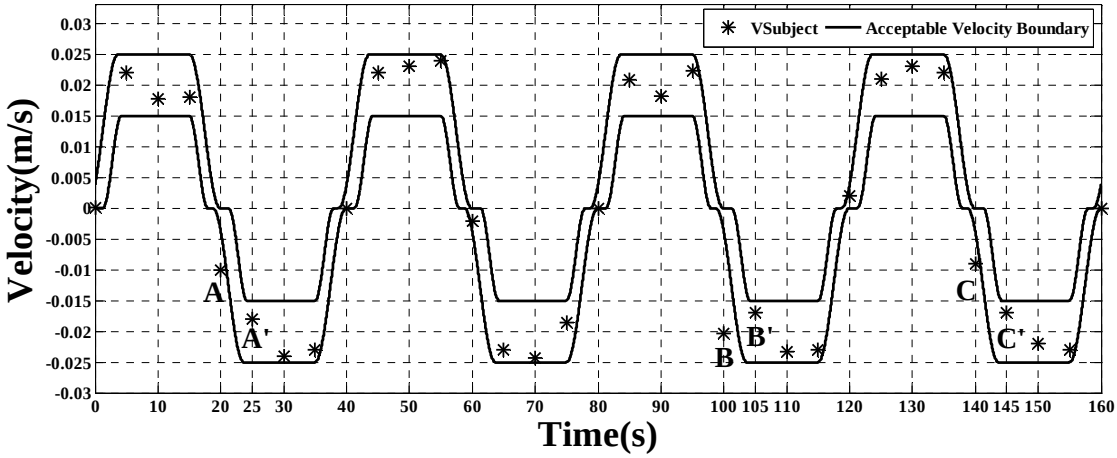


Figure 6 Calculated Average Velocities

We further analysed the amount of time taken by the low-level assistive controller (t_s, in seconds) to take $\dot{x}$ into the desired velocity range. Here t_s is defined as the settling time, which is the time taken between the moment the low-level assistive controller was activated and the actual velocity reached the boundary of the desired velocity range. The mean and standard deviation of t_s for all participants' data are presented in Table 3. It can be seen that the low-level assistive controller could assist the participants when needed and could quickly (i.e., in approximately *0.55* seconds) bring the participants' velocity within the desired range.

Table 3 Settling Time

Participant	Mean	Standard Deviation
P1	0.4723	0.1502
P2	0.5801	0.1937
P3	0.4929	0.1272
P4	0.545	0.232

6.2.2. High-Level Controller Evaluation

In order to demonstrate the efficacy of the proposed high-level controller, we had designed two experiments. In the first experiment, we had demonstrated the efficacy of the proposed high-level controller to modify the task when the participant improved his/her movement ability to track the desired trajectory. In the second experiment, we had demonstrated the efficacy of the high-level controller to modify the task in order to ensure the safety of the participants.

In the first experiment, we had used P1's low-level assistive controller results. Δp , which is the percentage change of velocity in (16), was selected as *30*, which could be varied based on participant's progress and the therapist's choice. It was observed from Table 2 that n_{10}=*8* and n_{50}=*3* and (16) was satisfied, thus *E1* was triggered and the plant symbol $\tilde{x}_1$ was generated from the interface. *difficult* $(\tilde{p}_2)$ state became active and the control symbol $\tilde{r}_2$ was generated. The interface converted this control symbol to $\alpha = 1$ and $\beta = 1$. Amount of the increment *delta* to increase the difficulty level of the task was an important issue that needed to be decided. In rehabilitation therapies, increasing $\dot{x}_d$ with a small increment would be more desirable especially for low-functioning stroke patients. In this experiment, we had incremented $\dot{x}_d$ by *20%*, where $delta = 0.004$. New desired velocity was calculated using (17), which was *0.024m/s*. The velocity boundaries were calculated using (11) as *0.03m/s* and *0.018m/s* for $\dot{x}_u$ and $\dot{x}_l$, respectively. We had asked P1 to perform the tracking task 50 times with this new velocity boundary. Low-level assistive controller provided robotic assistance to the participant as and when they were out of the new velocity band. It was observed that the P1 needed more robotic assistance when the desired velocity to complete the task was increased. It could be seen that P1 learned how to accomplish the task with practice (Table 4**).**

Table 4 Number of Times Robot Assisted for P1 with New Velocity Boundary

Trial Range / Number of Assistance for	1-10	11-20	21-30	31-40	41-50
P1	11	10	9	8	7

In the second experiment, we had assumed a safety event had occurred when P1 was performing the task with new increased velocity band. In this experiment, at some point of time during the task P1 wanted to pause for a while and then reset the pause button when she was ready to complete the rest of the task. This scenario might represent when a stroke patient want to pause for a while due to some discomfort. When the task had initially started, *E1* was triggered and the plant symbol $\tilde{x}_1$ was generated from the interface. *difficult* $(\tilde{p}_2)$ state became active and the control symbol $\tilde{r}_2$ was given to the interface. The interface converted this control symbol to constant values $\alpha = 1$ and $\beta = 1$. The plant equation (17) was used to calculate $\dot{x}_{dm}$ (the desired velocity), which was *0.024m/s*. The reaching task required participant to move *0.3m*, thus, the initial position (*0*), desired position (*0.3*) and desired $\dot{x}_{dm}$ (*0.024m/s*) was provided to the reference block to generate the smooth desired velocity trajectory from *A* to *B* (Fig.7-left-solid line).

When P1 pressed the pause button at *B*, *E7* was triggered. When *E7* was triggered, plant symbol $\tilde{x}_4$ was generated from the interface and *stop* $(\tilde{p}_4)$ state became active. When *stop* state was active, the high-level controller provided the control symbol $\tilde{r}_4$ and $\beta = 0$ was given to (17) and $\dot{x}_{dm}$ was determined as zero. The zero velocity could cause a sudden stop. In order to prevent P1 from suddenly stopping, the reference generator block was used to provide a smooth velocity trajectory to bring the motion to stop. In this case, the velocity was detected when *E7* was triggered and the desired velocity was given as zero and using the reference generator block, the smooth desired velocity was given to the low-level assistive controller from *B* to *C* (Fig.7-left-solid line). It could be seen that P1's position (Fig. 7 - right) did not change after the velocity became zero until P1 reset the pause button. *SE1* was set to *1* because the state was *difficult* and *E7*=1. When the participant reset the pause button, *E8* was triggered and $\tilde{x}_5$ plant symbol was given to the interface, and *pause* $(\tilde{p}_5)$ state became active

and the high-level controller provided $\tilde{r}_5$. Then $\tilde{x}_6$ was given to the interface because *SE=1*. The corresponding control symbol $\tilde{r}_2$ was generated, and $\alpha = 1$ and $\beta = 1$ values were given to (17) to calculate $\dot{x}_{dm}$, which was *0.024m/s*. It could be seen that the high-level controller resumed the task in such a manner that the participant could continue with the therapy with the same task parameters. The participant's position at the time of the triggering of *E8* was automatically detected and was given as an initial position to the reference generator block and the desired position was set to *0.3*. The velocity trajectory from *C* to *D* was generated and given to the low-level assistive controller (Fig.7-left-solid line). On the other hand, if we did not use this high-level controller, the desired velocity trajectory would not have been automatically modified to register the intention of the participant to pause the task. As a result, the velocity trajectory would have followed the dashed line in Fig. 7-left. In such a case, when P1 wanted to start the task again, the desired velocity trajectory would start at point *C'* with non-zero velocity (Fig.7-left-dashed line). This could create unsafe operating condition. In addition, since the desired velocity computation would not have included the pause action, restarting the task at point *C'* would not allow the completion of the task as desired. For example, in this case, if P1 had used the dashed velocity trajectory, she would start moving in the opposite direction at point *C'*. It could be possible to pre-program all types of desired velocity trajectories beforehand and retrieve them as needed. However, for non-trivial tasks such a mechanism might be too difficult to manage and extend as needed. The presented high-level controller provides a systematic mechanism to tackle such issues. It could also be seen that new velocity trajectories could be created dynamically using the generator block. In order to generate the required trajectories, the task parameters were needed. High-level controller monitored the progress of the task and made decision on the modification of the task parameters.

When the participant reached the desired position, which was *0.3m*, then the velocity trajectory from *D* to *E* was generated and given to the low-level assistive controller (Fig.7-left-solid line) so that P1 moved back to the starting position (Fig.7-right).

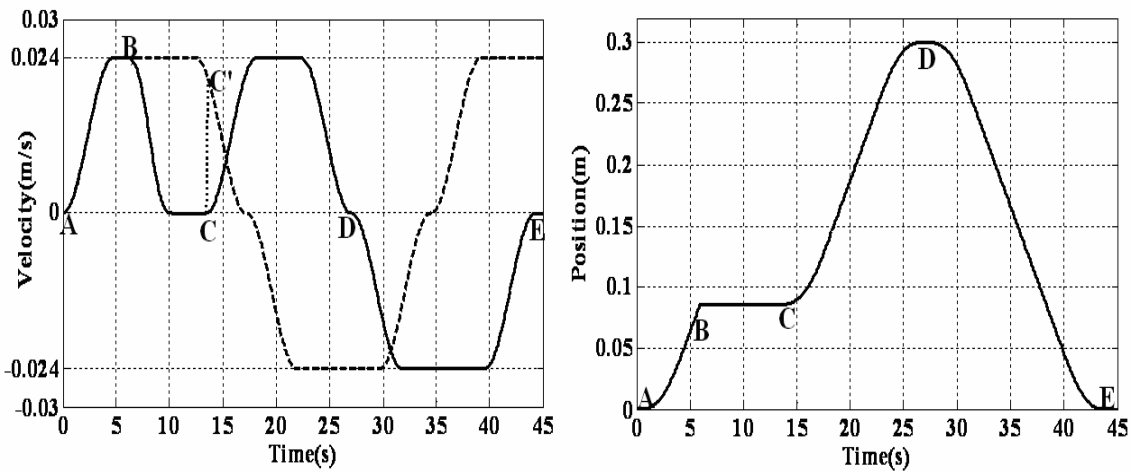


Figure 7 Motion Trajectories When Task is paused

As could be seen from the results of Experiments 1 and 2, the high-level controller determined the task parameters dynamically based on the participant's performance and monitored the safety related events to generate the necessary motion trajectories at the required time.

7. Conclusion and Future Work

In this paper we present a new control approach to offer robotic assistance for stroke patients that include the coordination between a high-level controller and a low-level assistive controller. The control architecture presented here is an example of a hybrid control system. There has been no work to our knowledge on designing similar type of control architecture for rehabilitation purposes. We present the design of a low-level assistive controller which is used to provide robotic assistance for an upper arm rehabilitation task. The high-level controller coordinates with the low-level assistive controller to improve the robotic assistance with the following objectives: 1) to monitor the upper arm rehabilitation task; and ii) to make necessary decisions to address the status of the task. We present a systematic design procedure for the high-level controller to accomplish the above objectives. Note that the proposed high-level controller can be integrated with other low-level controllers with minor modifications.

We have conducted experiments with unimpaired participants and demonstrated the usefulness of the high-level and low-level assistive controllers. The results of the use of the low-level assistive controller have demonstrated that the participants needed less assistance from the robot as they practiced more, which implies that the participant's ability to complete the desired motion within a defined velocity range have been improved. We have also

demonstrated that the low-level assistive controller provides assistance to the participants as and when needed and quickly brought the participant's velocity in the desired range (i.e., in approximately *0.55* seconds). The results of the use of the high-level controller have demonstrated that the task parameters could be determined dynamically based on the participant's performance and monitored for safety related events to generate the necessary motion trajectories at the required time. The speed of motion is used as the task parameter in this paper. However, the high-level controller can determine other task parameters such as desired reaching position. In some of the rehabilitation tasks, the reaching task is shaped by defining the target position closer to or away from the patient to change the difficulty level of the task. In such a case, for example, the high-level controller can determine the target position based on the participant's progress while monitoring the safety related events.

An important direction for future development involves testing the usability of the proposed control architecture with stroke patients. New methods to detect human state information can be integrated into the control architecture such as ECG signals can be used to monitor patients' heart rate to detect their exhaustion and a voice recognition system can be integrated to examine the patient's verbal commands. The proposed control architecture is flexible and extendible in the sense that new events can be included and detected by simply monitoring the additional state information from the human and the robot. In this regard, we are currently working with Vanderbilt University's Stallworth Rehabilitation Hospital to include additional human and robot information.

Acknowledgment

We gratefully acknowledge the help of Dr. Thomas E. Groomes who is the Medical Director of Spinal Cord and Traumatic Brain Injury Program, and therapist Sheila Davy of Vanderbilt University's Stallworth Rehabilitation Hospital for their feedback about experimental setup and task design during this work. The work was supported by Vanderbilt University Discovery grant.

References

[1] Matchar D.B., Duncan P.W.: Cost of stroke, Stroke Clin Updates, 5, 9-12 (1994).

[2] American Heart Association: Heart and Stroke Statistical Update, http://www.Americanheart.org/statistics/stroke.htm (2006).

[3] Taub E.: Harnessing Brain Plasticity through Behavioral Techniques to Produce New Treatments in Neurorehabilitation. American Psychologist, 59(8), 692-704 (2004).

[4] Krebs H. I., Hogan N., Aisen M.L., Volpe B.T.: Robot–aided neurorehabilitation. IEEE Trans. on Rehab. Eng. 6, 75-87 (1999).

[5] Kahn L.E; Lum P.S.; Rymer W.Z; Reinkensmeyer D.J.: Robot-assisted movement training for the stroke-impaired arm: Does it matter what the robot does? J. of Rehab. Research & Development, 43(5), 619-630 (2006).

[6] Lum, P.S.; Burgar, C.G.; Van der Loos, H.F.M.; Shor, P.C.; Majmundar M.; Yap R.: MIME robotic device for upper-limb neurorehabilitation in subacute stroke subjects: A follow-up study. J. of Rehab. Research & Development. 43(5), 631-642 (2006).

[7] Krebs H.I., Palazzolo J.J., Dipietro L., Ferraro M., Krol J., Rannekleiv K. , Volpe B.T. , Hogan N.: Rehabilitation Robotics: Performance-Based Progressive Robot-Assisted Therapy. Autonomous Robots. 15(1), 7-20 (2003).

[8] Krebs H. I, Ferraro M., Buerger S.P, Newbery M. J., Makiyama A., Sandmann M., Lynch D., Volpe B. T., Hogan N.:Rehabilitation robotics: pilot trial of a spatial extension for MIT-Manus. Journal of NeuroEngineering and Rehabilitation. 1(5), 1-15, (2004).

[9] Kahn L.E., Zygman M.L.Rymer W.Z, Reinkensmeyer D.J: Robot-assisted reaching exercise promotes arm movement recovery in chronic hemiparetic stroke: a randomized controlled pilot study. Journal of NeuroEngineering and Rehabilitation. 3(12), 1-13 (2006).

[10] Loureiro R., Amirabdollahian F., Topping M., Driessen B., Harwin W.:Upper limb mediated stroke therapy - GENTLE/s approach. Autonomous Robots. 15, 35-51 (2003).

[11] Taub, E., Lum, P.S., Hardin, P., Mark V.W. & Uswatte, G. AutoCITE: Automated Delivery of CI Therapy With Reduced Effort by Therapists. Stroke. (36), pp. 1301-1304 (2005).

[12] Koutsoukos X.D., Antsaklis P.J., Stiver J.A., Lemmon M.D.: Supervisory control of hybrid systems. Proceedings of the IEEE on Special Issue on Hybrid Systems: Theory and Applications, 88(7), 1026 – 1049 (2000).

[13] Antsaklis P.J., Koutsoukos X.D.: Hybrid Systems: Review and Recent Progress. Chapter in Software-Enabled Control: Information Technologies for Dynamical Systems, T. Samad and G. Balas, Eds., IEEE Press. 1-29 (2003).

[14] Antsaklis P.J.: A Brief Introduction to the Theory and Applications of Hybrid Systems. Proceedings of the IEEE on Special Issue on Hybrid Systems: Theory and Applications, 88(7), 879-887 (2000).

[15] PUMA 560 Related Sites on the Internet, http://www.ee.ualberta.ca/~jasmith/puma/pumasites.html.

[16] Armstrong B., Khatib O., Burdick J.:The Explicit Dynamic Model and Inertial Parameters of the PUMA 560 Arm. IEEE Intl. Conf. on Robotics and Automation, 510-518, 1986.

[17] Craig J. J.:Introduction to Robotics Mechanics and Control.2nd Edition. Addison-Wesley (1986).

[18] Paul R., Shimano B., Mayer G. E.: Kinematic Control Equations for Simple Manipulators. IEEE Transactions on Systems, Man and Cybernetics, 11(6), 449-455 (1981).

[19] Erol D., Sarkar N.: Design and Implementation of an Assistive Controller for Rehabilitation Robotic Systems. Inter. J. of Adv. Rob. Sys. 4(3), (2007).

[20] Culmer, P., Jackson, A. ,Richardson, R., Bhakta, B., Levesley, M., Cozens, A.: An admittance control scheme for a robotic upper-limb stroke rehabilitation system. Int. Conf. On Engineering in Medicine and Biology Society, 5081 – 5084 (2005).

[21] Sciavicco L., Siciliano B.: Modeling and Control of Robot Manipulators. McGrawHill, (1996).

[22] Mallapragada V., Erol D., Sarkar N.: A New Method for Force Control for Unknown Environments. IEEE/RSJ International Conference on Intelligent Robots and Systems, 4509-4514 (2006).

[23] Stateflow, Mathworks Inc, http://www.mathworks.com/products/stateflow/?BB=1.

CHAPTER IV: MANUSCRIPT 3

AN APPROACH TO SMOOTH HUMAN-ROBOT INTERACTION IN ROBOT-ASSISTED REHABILITATION

Duygun Erol & Nilanjan Sarkar

(Submitted to International Journal of Robotics Research)

Abstract

The goal of this work is to develop a control framework to provide robotic assistance for rehabilitation tasks to the subjects in such a manner that the interaction between the subject and the robot is smooth. This is achieved by designing a methodology that automatically adjusts the control gains of the robot controller to modify the interaction dynamics between the robot and the subject. In order to automatically determine the control gains for each subject, an artificial neural network (ANN) based proportional-integral (PI) gain scheduling controller is proposed. The human arm model is integrated within the controller where the ANN uses estimated human arm parameters to select the appropriate PI gains for each subject such that the resultant interaction dynamics between the subject and the robot could result in smooth interaction. Experimental results are presented to demonstrate the efficacy of the proposed ANN-based PI gain scheduling controller on unimpaired subjects.

KEY WORDS – rehabilitation robotics, human arm parameter estimation, gain scheduling, smooth interaction

1. Introduction

Stroke is a highly prevalent condition [1], especially among the elderly that results in high costs to the individual and society [2]. According to the American Heart Association (2006), in the U.S., approximately 700,000 people suffer a first or recurrent stroke each year [1]. It is a leading cause of disability, commonly involving deficits of motor function. Recent clinical results have indicated that movement assisted therapy can have a significant beneficial impact on a large segment of the population affected by stroke or other motor deficit disorders. Experimental evidence suggests that intensive movement training of new motor tasks is required to induce long-term brain plasticity. New techniques adopting a task-oriented approach have been developed to encourage active training of the affected limb, which

assume that control of movement is organized around goal-directed functional tasks [3]. For example, “Shaping” is one of the task-oriented behavioural training techniques employed in Constraint-Induced Movement Therapy (CIMT), [4] which has the effect of placing optimal adaptive task practice procedures into a systematic, standardized, and quantified format. The availability of such training techniques, however, is limited by the amount of costly therapist’s time they involve, and the ability of the therapist to provide controlled, quantifiable and repeatable assistance to complex arm movement. Consequently, robot assisted rehabilitation that can quantitatively monitor and adapt to patient progress, and ensure consistency during rehabilitation may provide a solution to these problems.

Robot-assisted rehabilitation for rehabilitation of the stroke patients has been an active research area, which provide repetitive movement exercise and standardized delivery of therapy with the potential of enhancing quantification of the therapeutic process [5]-[12]. The first robotic assistive device used as a therapeutic tool, the MIT Manus [9][10] uses an impedance controller to provide assistance to move patient’s arm to the target position in an active assisted mode, where patients can visually see their movement and target location. Mirror Image Movement Enabler (MIME) [5] uses a PUMA 560 manipulator to provide assistance to move the subject’s arm with a pre-programmed position trajectory using Proportional-Integral-Derivative (PID) controller. Assisted Rehabilitation and Measurement (ARM) Guide [8],[12] is another robotic system that is capable of generating both horizontal and vertical motion, which provides assistance or resistance to the patient’s movement to complete the reaching task. The GENTLE/s is a commercially available haptic robot used to provide assistance to patients to move to the target positions along with a predefined path using admittance control. The subject’s movement trajectory is represented in the virtual environment [11]. Studies with these robotic devices verified that robot-assisted rehabilitation results in improved performance of functional tasks for stroke patients. New rehabilitation therapy environments are under development to permit training of real-life functional tasks involving reach, grasp, as well as object transportation in three dimensional (3-D) space [6] [7] for stroke patients.

In robot-assisted therapies, patients attempt to make voluntary movements while the robotic device provides assistance to complete the desired rehabilitation task. It is desirable to provide the robotic assistance to patients in such a manner that the resulting interaction between the robot and the patient is smooth. In the literature jerk has been used as a measure of smoothness of the prescribed motion, where minimum jerk implied smooth movement [13]- [26]. The numerical value of jerk has been used as a metric to determine the movement

smoothness during stroke recovery [27]. Furthermore, smoothness of the movement has been quantified by the mean squared magnitude of jerk [13]-[26],[28]. Many of the existing robotic rehabilitation devices such as [8]-[12] try to ensure the smoothness of motion by specifying the desired trajectory of the task as the minimum jerk trajectory as originally proposed by [13]. The idea is that if the subject can follow the desired minimum jerk trajectory, the motion will be smooth and consequently, the force applied by the subject will also be smooth. Here the smoothness is measured by the rate of the change of the force applied by the subject. However, when the subject cannot follow the specified desired motion trajectory entirely by his/her own effort, he/she will need robotic assistance. Note that the manner in which this robotic assistance is imparted could affect the rate of change of the subject applied force. For example, if the robot provides assistance with a high overshoot, then the subject is required to overcome this overshoot by changing his/her applied force in a rapid manner. This scenario could result in more variation in subject applied force, which implies non-smooth interaction. In here, we use the rate of change of the applied force by the subject as a measure of smooth interaction, which is termed as smooth interaction index or *SII*.

In this work, we will show that suitable modification of the interaction dynamics between the robot and the subject will result in less variation of the force applied by the subject (smooth interaction) to complete the task for any type of desired trajectory including the minimum jerk trajectory. The interaction dynamics depends on the dynamics of the human arm as well as the dynamics of the robot. While the dynamics of the human arm cannot be modified, the dynamics of the robot can be changed by adjusting the control parameters of the robot controller. A control framework is proposed in here, which is called an artificial neural network (ANN) based Proportional-Integral (PI) gain scheduling controller, that will automatically adjust the control gains for each subject such that the resultant interaction dynamics between the subject and the robot could result in smooth interaction. The control gains are determined based on online estimation of the human arm parameters (i.e., stiffness). The online estimation is performed in parallel with the control computation, and thus does not introduce any destabilizing effect on the control performance. The ANN, which is trained offline, adjusts the control gains online. The proposed controller combines the benefit of system identification technique with the robustness of neural network-based methods.

This paper is organized as follows: Section 2 presents the control approach that includes the basic assistive controller, the human arm parameter estimation technique, the design of the gain scheduling to result in smooth interaction, and the gain switching mechanism.

Section 3 describes the experimental setup, tasks and experimental procedure. In Section 4, we present the results obtained with the real-time implementation of the proposed ANN-based PI gain scheduling controller on unimpaired subjects. Section 5 presents a brief discussion on the performance of the proposed ANN-based PI gain scheduling controller and possible future research directions.

2. Control Framework

In our previous work, we propose an assistive controller to provide robotic assistance to the subjects to complete a rehabilitation task [17]. An outer force feedback loop is designed around an inner position loop. The tracking of the reference trajectory is guaranteed by the inner motion control. Previous research in rehabilitation robotics mostly employed a position based Proportional-Integral-Derivative (PID) control, an impedance control and an admittance control [5]-[12]. The assistive controller is similar to an impedance controller; however it allows specifying the desired time varying force directly.

In this work, we augment the basic assistive controller in such a manner that the resulting interaction between the robot and the subject is smooth. It will be shown in this work (Section 4) that different control gains result in different smooth interaction index (*SII*) for the same subject when performing the same task. Furthermore, same control gains may result in different *SII* for different subjects. As a result, one needs to choose suitable control gains that result in smooth interaction for each subject. Thus, it would be ideal if the control gains could be mapped directly to *SII*. However, it could also be possible to obtain similar SII characteristics using different gains for different subjects. This necessitates choosing the control gains that can differentiate each individual subject for the purpose of the smooth interaction. Since the human arm characteristics (e.g., stiffness) could vary for different subjects as well as for the same subject during various phases of the task, we propose to use human arm parameters to differentiate subjects as well as to map the suitable control gains subject to the smooth interaction. The mapping between the human arm parameters and the control gains is nonlinear. As a result, we take advantage of artificial neural network (ANN) for mapping, which has the ability to map complex nonlinear relationships between input and output data using patterns learned during training. In order to develop and use this map, ANN requires the knowledge of the human arm parameters, which is determined using a system identification method. The ANN is first trained offline using a training data set consisting of human arm parameters as inputs and suitable Proportional-Integral (PI) control gains that result in smooth interaction as outputs. Afterwards, ANN is used in conjunction with the

human arm parameter estimation module to predict the suitable gains for the subject based on the estimated human arm parameters in real-time. The online estimation and the gain prediction using ANN are performed in parallel with the control computation, and thus they do not introduce any destabilizing effect on the control performance. However, it should be noted that instantaneous switching of control gains could destabilize the system or cause a large overshoot. Hence, a gain switching mechanism is designed to achieve a smooth transfer of control gains.

Fig. 1 shows the schematic of the overall control framework. While the subject performs the task, his/her arm characteristics are determined using the human arm parameter estimation module. The output of the human arm parameter estimation module is given to the gain prediction to result in smooth interaction module as an input to predict the suitable control gains. The above-mentioned human arm parameter estimation and gain prediction, as shown in a dashed box in Fig. 1, can be done at the same sampling frequency as the robot controller or at a slower rate depending on the computational capabilities of the system. It may not be needed to estimate and change the control gains frequently because the human arm parameters may not change rapidly. Note that the robot controller does not require that this parameter estimation and gain prediction to be completed at the same sampling time. In the presented work, we perform all the computation shown in Fig. 1 at the same sampling time (1000 Hz). However, the gains were changed only when there was substantial difference between the current and the predicted gains. The gain switching module ensures that when the control gains are changed, the system does not experience instability.

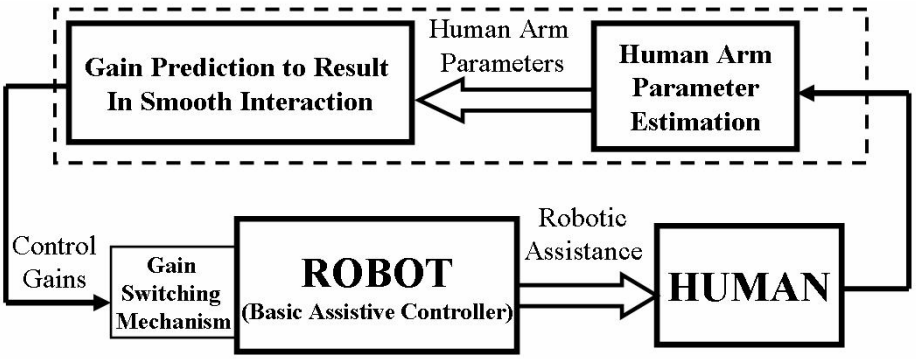


Fig. 1. Control Framework

In what follows, we first present the basic assistive controller, the human arm parameter estimation technique, design of the gain prediction to result in smooth interaction, and the gain switching mechanism.

2.1. Basic Assistive Controller

The existing robotic rehabilitation systems operate in robot task-space to provide robotic assistance to the patients to follow a desired trajectory to complete a rehabilitation task [5]-[12]. Recently, a human-arm joint impedance controller is proposed, which operates in joint-space, to provide assistance to the subjects to follow desired joint angle trajectory [18] specified for each individual joint (e.g., elbow joint). It is still not clear, however, whether the assistance in the task-space or in the joint-space will likely to have the best results for rehabilitation purposes. In this work, we design a controller that is responsible for providing the robotic assistance to subjects to complete a rehabilitation task in task-space [17]. In this controller, an outer force feedback loop is designed around an inner position loop (Fig. 2). The tracking of the reference trajectory is guaranteed by the inner motion control [19]. The desired force, which is given as a force reference to the controller, is computed by a planner. The proposed controller is similar to an impedance controller; however it allows specifying the reference time varying force directly.

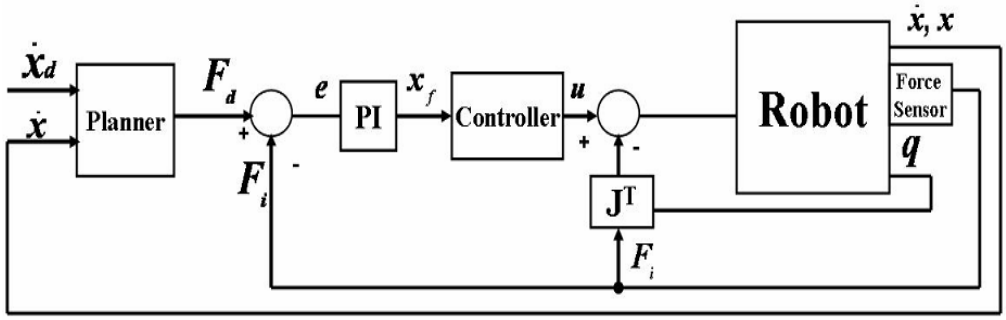


Fig. 2. Basic Assistive Controller

The dynamic equations of motion for the robot are given by:

$$\begin{aligned} \Gamma &= M(q)\ddot{q} + Co(q)(q,\dot{q}) + Ce(q)\left|\dot{q}^2\right| + G(q) \\ u - J^T(q)F &= M(q)\ddot{q} + V(q,\dot{q}) + G(q) \end{aligned} \tag{1}$$

where $M(q)$ represents the inertia matrix, $V(q,\dot{q})$ is the summation of the matrix of coriolis torques $Co(q)(q,\dot{q})$ and centrifugal torques $Ce(q)\left|\dot{q}^2\right|$, $G(q)$ is the vector of gravity torques. Γ is the generalized joint force torque which is calculated using $u - J^T(q)F$, where u is the input to the manipulator, $J(q)$ is the Jacobian matrix and F is the contact force exerted by the manipulator. Using inverse dynamics control, manipulator dynamics are linearized and decoupled via a feedback. The dynamic equation of the robotic manipulator was given in (1). Control input u to the manipulator is designed as follows:

$$u = M(q)y + V(q,\dot{q}) + G(q) + J^T F \tag{2}$$

which leads to the system of double integrators

$$\ddot{q} = y \tag{3}$$

In (3), y represents a new input. The new control input y is designed so as to allow tracking of the desired force F_d. To this purpose, the control law is selected as follows:

$$y = J(q)^{-1} M_d^{-1}(-K_d \dot{x} + K_p(x_f - x) - M_d \dot{J}(q,\dot{q})\dot{q}) \tag{4}$$

where x_f is a suitable reference to be related to force error. M_d (mass), K_d (damping) and K_p (stiffness) matrices specify the target impedance of the robot. x and $\dot{x}$ are the position and velocity of the end-effector in the Cartesian coordinates, respectively. The relationship between the joint space and the Cartesian space acceleration is used to determine position control equation.

$$\ddot{x} = J(q)\ddot{q} + \dot{J}(q,\dot{q})\dot{q} \quad and \quad \ddot{x} = J(q)y + \dot{J}(q,\dot{q})\dot{q} \tag{5}$$

By substituting (4) into (5), we obtain

$$\ddot{x} = J(q)(J(q)^{-1} M_d^{-1}(-K_d \dot{x} + K_p(x_f - x) - M_d \dot{J}(q,\dot{q})\dot{q})) + \dot{J}(q,\dot{q})\dot{q}$$

$$\ddot{x} = -M_d^{-1} K_d \dot{x} + M_d^{-1} K_p (x_f - x)$$

$$M_d \ddot{x} + K_d \dot{x} + K_p x = K_p x_f \tag{6}$$

Equation (6) shows the position control tracking of x with dynamics specified by the choices of K_d, K_p and M_d matrices. Impedance is attributed to a mechanical system characterized by these matrices that allows specifying the dynamic behavior. Let F_d be the desired force reference, which is computed using a PID velocity loop:

$$F_d = P_d(\dot{x}_d - \dot{x}) + I_d \int(\dot{x}_d - \dot{x})dt + D_d \frac{d(\dot{x}_d - \dot{x})}{dt} \tag{7}$$

where $\dot{x}_d$, $\dot{x}$, P_d, I_d and D_d are the desired velocity, actual velocity, the proportional, integral and derivative gains of the PID velocity loop, respectively. The relationship between x_f and the force error is expressed in (8) as:

$$x_f = P(F_d - F_i) + I \int(F_d - F_i)dt \tag{8}$$

where P and I are the proportional and integral gains, respectively, and F_i is the force

applied by the human. Equations (6) and (8) are combined to obtain below equation:

$$M_d\ddot{x} + K_d\dot{x} + K_p x = K_p(P(F_d - F_i) + I\int(F_d - F_i)dt) \quad (9)$$

We can observe from (9) that the desired force response is achieved by controlling the position of the manipulator.

2.2. Human Arm Parameter Estimation

We propose to use human arm parameters to differentiate subjects for mapping the suitable control gains to result in smooth interaction. A human arm can be characterized by its impedance parameters (mass, stiffness and damping). Many researchers in the fields of biology and robotics have developed models of the impedance characteristics of the human arm [20]-[27]. It has been shown that human arm stiffness varied greatly between subjects, tasks, perturbation patterns and experimental devices [20]-[23], [26].

We use a rehabilitation task that requires low velocity since high velocity task is not feasible for most low-functioning stroke patients. It was previously discussed in [26] that in the low velocity range (0.02m/s or less); the stiffness coefficient dominates over the mass and the damping coefficient during a human-robot task. Thus we use the following single degree of freedom arm dynamic model:

$$\Delta F = K\Delta x \quad (10)$$

where K is the stiffness of the human arm in the task space. The above model uses equilibrium point hypothesis [28]-[32]. Equilibrium point hypothesis suggests that movement of the subject's arm is generated by shifting the equilibrium position of the arm at a constant rate from its initial configuration to final configuration. In (10) Δx is difference between the current arm position and the equilibrium position, and ΔF is the difference between the force applied by the subject at the current arm position and the force applied by the subject at the equilibrium position.

Various system identification techniques can be used to estimate the stiffness in (10). In this work ARX (Auto Regressive eXogenous) model is chosen to estimate the stiffness of the human arm [33]. ARX model structure is one of the simplest parametric structures one can use with very little numerical difficulty. The parameters of the ARX model are estimated using a recursive least-squares (RLS) method. RLS method is one of the most well known algorithms used in adaptive filtering, system identification and adaptive control. Its popularity is mainly due to its fast convergence speed [34].

Let n be the time-step index. The following difference equation is obtained:

$$\Delta F[n] = K\Delta x[n] \tag{11}$$

Equation (11) can be cast into the regressor form as follows:

$$y[n] = \varphi^T[n]\theta[n] \tag{12}$$

with $\theta[n] = K[n]$ representing the parameter matrix, $\varphi[n] = \Delta x[n]$ the regression vector, and $y[n] = \Delta F[n]$ the output vector. The RLS solution to determine K is given by:

$$\theta[n] = \theta[n-1] + G[n](y[n] - \varphi^T[n]\theta[n-1]) \tag{13}$$

where the gain factor $G[n]$ determines how the current prediction error $y[n] - \varphi^T[n]\theta[n-1]$ affects the update of parameter estimation. $G[n]$ is determined using:

$$G[n] = \frac{C[n-1]\varphi[n]}{\lambda + \varphi^T[n]C[n-1]\varphi[n]} \tag{14}$$

where λ is the forgetting factor that influences the weight given to earlier data relative to the newly acquired data. $C[n]$ is the covariance matrix of the estimated parameter, which is calculated using Equation (14).

$$C[n] = \frac{1}{\lambda}\left[C[n-1] - \frac{C[n-1]\varphi[n]\varphi^T[n]C[n-1]}{\lambda + \varphi^T[n]C[n-1]\varphi[n]}\right] \tag{15}$$

The initial guess for the covariance matrix C and the forgetting factor λ are specified by the user to estimate K. For our application, λ is chosen to be *0.999*. The time taken by the RLS algorithm to converge to estimate K is dependant on the forgetting factor. A low forgetting factor quickens the convergence, but is more susceptible to noise whereas a higher forgetting factor takes longer time to converge but is less susceptible to noise.

2.3. Gain Prediction to Result in Smooth Interaction

In order to predict the gains to obtain smooth interaction, a mapping is designed using artificial neural network (ANN) between the human arm parameters and the suitable control gains. In this work, the human arm stiffness estimate K is used as the sole input vector and suitable Proportional (P) and Integral (I) gains that result in smooth interaction are used as the corresponding target (output) vectors to train the neural network offline. The most common form of supervised training is the error backpropagation algorithm. We use a multi-layer ANN with back-propagation method for mapping the arm stiffness to the suitable control gains. The training data set is chosen in such a way that it spans a large range of the stiffness to cover a variety of subjects. Once the ANN is trained offline, it can work online in

conjunction with the human arm parameters estimation module to generate suitable control gains based on estimated stiffness of the human arm.

2.4. Gain Switching Mechanism

The control gains are predicted using the ANN that has been trained offline. If the predicted gains are instantaneously switched during the execution of the task, it could destabilize the system or cause undesired transients in the force response, which is not preferred in rehabilitation therapies. Hence, to achieve a satisfactory force response, a smooth control signal has to be retained during switching. Therefore, gain switching can be viewed as a bumpless transfer between two PI controllers with different gains. A Bumpless transfer strategy for switching between a manual controller and an automatic PI controller is presented in [35]. In our previous work [36], bumpless switching between PI controllers with different gains to track a desired force for unknown environments was presented. The gain switching mechanism modifies the position reference, which is the input for the inner loop of the controller, at the time of the switching in such a way that it is equal to the position reference at the time before switching occurred. This mechanism ensures that the position reference is indeed continuous during switching which guarantees bumpless switching. The PI control action in (8) can be written as below:

$$x_f = Pe + I\int e dt \tag{16}$$

where e is the force error and calculated using $e = F_d - F_i$. (16) can be written as:

$$x_{fi}(t) = P_i\, e(t) + I_i\,(X_i(t) + X_{io}) \tag{17}$$

P and I are the proportional and integral gains, respectively, where $x_{fi}(t)$ is the position reference determined using the initial gains P_i and I_i. $X_i(t)$ represents the integral action and X_{io} is the initial condition of the error integrator. If t_s is the time of switching, then equation (17) can be used to find the position reference just before the time of switching as described below:

$$x_{fi}(t_s^-) = P_i\, e(t_s^-) + I_i\left(X_i(t_s^-) + X_{io}\right) \tag{18}$$

Let the gains predicted by the ANN be represented as P_a and I_a. The position reference just after switching is given as below:

$$x_{fa}(t_s^+) = P_a\, e(t_s^+) + I_a\left(X_a(t_s^+) + X_{ao}\right) \tag{19}$$

where $X_a(t)$ represents the integral action and X_{ao} is the initial condition of the error integrator associated with the ANN gains. Note that the force error does not change significantly during the switching procedure since the sample time for the controller is very small for small sampling time. Hence,

$$e(t_S^-) \approx e(t_S^+) \tag{20}$$

The integral action associated with the ANN gains is reset during switching so that,

$$X_a(t_S^+) = 0 \tag{21}$$

If the initial condition of the error integrator associated with the ANN gains is defined as:

$$X_{ao} = \frac{x_{fi}(t_S^-) - P_a e(t_S^-)}{I_a} \tag{22}$$

The relationships (20)-(22) are substituted into (19) to obtain:

$$x_{fi}(t_S^-) = x_{fa}(t_S^+) \tag{23}$$

Equation (23) ensures that the position reference is indeed continuous during switching which guarantees bumpless transfer.

Fig. 3 presents a detailed overall control architecture of the proposed ANN-based PI gain scheduling controller that has the ability to result in smooth interaction by adjusting the control gains. The human arm parameters are determined using the system identification module. The output of the system identification module is given to the artificial neural network (ANN) module as an input to predict suitable control gains to obtain smooth interaction. When the predicted control gains are changed, gain scheduling mechanism ensures the smooth transfer of control gains.

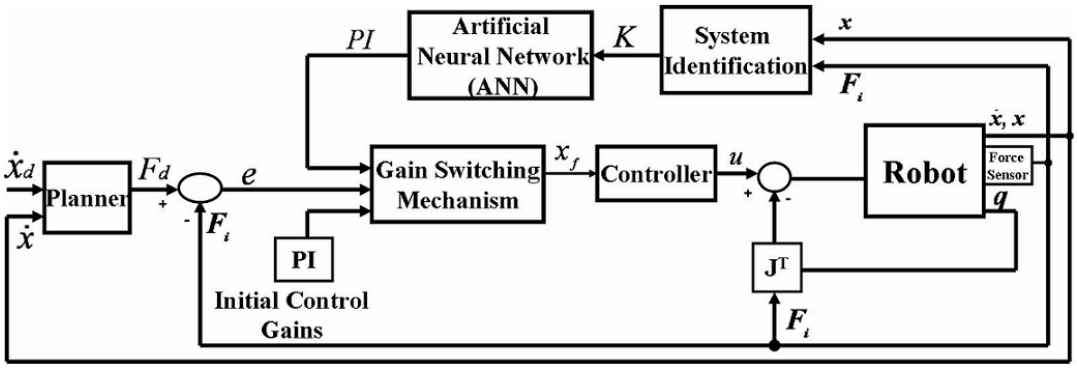


Fig. 3. ANN-based PI Gain Scheduling Control Architecture

3. Experiments and Tasks

In this section, we first present the rehabilitation robotic system that is designed to evaluate the proposed ANN-based PI gain scheduling controller. Then the descriptions of the tasks and experiments are introduced.

3.1. The Rehabilitation Robotic System

We have developed a rehabilitation robotic system that uses a PUMA 560 robotic manipulator, which is augmented with a force-torque sensor and a hand attachment device (Fig. 4). The microcontroller board of the PUMA is replaced to develop an open architecture system to allow implementing the proposed ANN-based PI gain scheduling controller. A tool frame is introduced to include the location of the human arm through the hand attachment device.

3.1.1. Hardware

The PUMA 560 is a 6 degrees-of-freedom (DOF) device consisting of six revolute joints. Each major joint (joints 1, 2 and 3) is equipped with an electromagnetic brake, which is activated when power is removed from the motors, thereby locking the robot arm in a fixed position. The technical specifications of this robotic device can be found in [37]. In order to record the force and torque, an ATI Gamma force/torque sensor [38] is used. The robot is interfaced with Matlab/Real-time Workshop to allow fast and easy system development. The force values recorded from the force/torque sensor are obtained using a National Instruments PCI-6031E data acquisition card with a sampling time of 0.001 seconds. The joint angles of the robot are measured using encoder with a sample time of 0.001 seconds from a Measurement Computing PCI-QUAD04 card. The torque output to the robot is provided by a Measurement Computing PCIM-DDA06/16 card with the same sample time. A computer monitor is placed in front of the subject to provide visual feedback about his/her motion trajectory during the execution of the task.

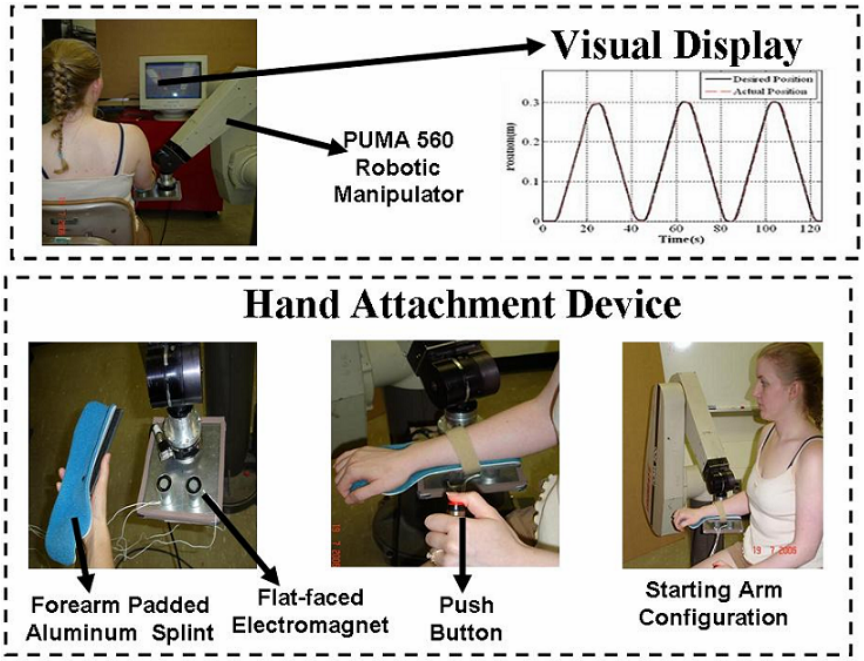


Fig. 4. Subject Arm attached to Robot

3.1.2. Hand Attachment Device

Since in this work we are primarily interested in effecting assistance to the upper arm, a hand attachment device is designed where the subject's arm is strapped into a splint that restricts wrist and hand movement. Typically, to practice rehabilitation tasks with robotic devices, the stroke survivor's impaired arm is attached to a robot, which supports the arm against gravity. The subject is asked to use the arm to make a movement with or without the assistance from the robot [5]-[12]. In here, a PUMA 560 is attached to a splint to provide assistance to the upper arm movement using the proposed ANN-based PI gain scheduling controller (Fig. 4). Forearm padded aluminum splint (from MooreMedical), which ensures the subject's comfort, is used as a splint in this device. A steel plate with proper grooves is designed that holds two small flat-faced electromagnets (from Magnetool Inc.) that are screwed on it. This plate is also screwed with the force-torque sensor, which provides a rigid connection with the robot. A light-weight steel plate under the splint is attached, which is then attached to the electromagnets of the plate. These electromagnets are rated for continuous duty cycle (100% duty cycle), i.e., they can run continuously at normal room temperature. Pull ratings of these magnets are 40lb. Two electromagnets have been used to have a larger pulling force to keep the splint attached to the hand attachment device. An automatic release (AU) rectifier controller (Magnetool Inc.) has been used to provide a quick release of these electromagnets. A push button, which has been connected to the AU Rectifier Controller, is used to magnetize and demagnetize the electromagnets when the subject wants

to remove the hand attachment device from the robotic manipulator in a safe and quick manner.

3.1.3. Safety discussion about both the use of a PUMA 560 Robotic Manipulator and hand attachment device

Ensuring safety of the subject is a very important issue when designing a rehabilitation robotic system. Thus, in case of emergency situations, therapist can press an emergency button. The patient and/or the therapist can quickly release the patient's arm from the PUMA 560 by using the quick-release hand attachment device (as described above) to deal with any physical safety related events. In order to release the subject's arm from the robot, the push button is used. When the push button is pressed electromagnets are demagnetized instantaneously and the subject is free to remove the splint from the robot. This push button can also be operated by a therapist. Additionally, the corner of the arm device has been covered with a foam self stick tape in order to avoid sharp surface.

3.2. Task Design

Let us first briefly review the task design of some well-known robotic rehabilitation systems. MIT Manus uses impedance controller to provide assistance to move the patient's arm to the target position, where patients can visually see their movement and target [9][10]. MIME provides assistance to move the subject's arm with a pre-programmed position trajectory using proportional-integral-derivative (PID) controller [5]. The subject is asked to maintain a specified off-axis force while he/she is trying to reach toward a goal position using ARM Guide [8],[12]. The GENTLE/s provides assistance to patients to move to the target positions along with a predefined path using admittance control. The subjects visually see their movement and the target in a virtual environment [11]. ADLER, which is a robot therapy environment, is developed to assist an impaired arm along trajectories for real-life tasks (such as reach, grasp in 3-D space) using admittance control [6] [7]. The therapy tasks designed for the existing rehabilitation robotic devices require predominantly shoulder motion or elbow motion, or some of them require the combination of both shoulder and elbow motion.

A reaching task is chosen that is commonly used for rehabilitation of upper extremity after stroke. In this task, the subjects are asked to move their arms in the forward direction to reach a desired point in space and then bring it back to the starting position repeatedly within

a specified time. In other words, they have to follow a desired position trajectory. The desired position trajectory in here is selected as the minimum-jerk trajectory [13] which is given as:

$$x(t) = x_i + (x_f - x_i)\left(10\left(\frac{t}{d}\right)^3 - 15\left(\frac{t}{d}\right)^4 + 6\left(\frac{t}{d}\right)^5\right) \quad (24)$$

where x_i, x_f and d are initial position, final position and duration of the movement, respectively. The subjects are required to follow the tip of the desired motion trajectory which represents the desired velocity. The reaching task designed in here requires combination of the shoulder and elbow which could increase the active range of motion (AROM) in shoulder and elbow in preparation of later functional reaching activities in rehabilitation. The allowable motion is restricted only to the direction of the task. For example, if the task requires the subject to move his/her arm in the Y-direction, then he/she is not allowed to move his/her arm in X or Z directions. However, he/she can move his/her arm in the Y-direction at a velocity that could be the same, higher or lower than the desired velocity. Although, in this work the motion of the arm is constrained in the horizontal plane in one direction (along the Y-axis), it could also be designed for other directions (e.g., X-axis) or combination of directions (e.g. XY-axes) based on task requirements (only shoulder or elbow motion or the combination of shoulder and elbow motion).

We have designed this reaching task such that the subjects not only make repetitive movement but also pay attention to the desired motion from visual feedback. The task was designed in such a manner that it required cognitive processing. Including cognitive processing in the task design is an important criterion because it had been previously shown that the movement tracking task that requires cognitive processing achieved greater gains for brain reorganization of stroke patients than that of movement task that does not require cognitive processing [39],[40]. In order to include cognitive processing within this reaching task, the subjects are asked to follow a visually presented desired motion trajectory that is likely to command their concentration. The subjects receive visual feedback of both their actual position and the desired position trajectories on a computer screen, which is placed in front of them. They are asked to pay attention to tracking the desired position trajectory as accurately as possible, which keeps them focused on the task. The visual feedback is used not only to inform the subjects of how closely they are tracking the desired motion but also as a motivational factor to keep them focused on the task.

3.3. Experiment Procedure

Subjects are seated in a height adjusted chair as shown in Fig. 4 (top left) during the performance of the task. The height of the PUMA 560 robotic manipulator has been adjusted for each subject to start the tracking task in the same arm configuration. The starting arm configuration is selected as shoulder at neutral position 0° and elbow at 90° flexion position. The task requires moving the arm in forward flexion to approximately 60° in conjunction with elbow extension to approximately 0°. Subjects are asked to place their forearm on the hand attachment device as shown in Fig. 4 (bottom left), when the starting arm configuration is fixed. The push button has been given to the subjects that can be used during the task execution in case of emergency situations (Fig. 4- bottom middle). The subjects receive visual feedback of their position on a computer monitor on top of the desired position trajectory (Fig. 4-top right).

3.4. Description of Experiments

Two experiments were conducted in this research: i) to demonstrate each subject required different control gains to result in smooth interaction during the execution of the rehabilitation task, and ii) to evaluate the proposed ANN-based PI gain scheduling controller. Experiments were performed on unimpaired subjects.

Experiment 1: The purpose of this experiment is to demonstrate that each subject may require different control gains for the same task in order to result in smooth interaction.

10 subjects were asked to perform the tracking task that was described in Section 3.2. Four female and six male, 20-32 years old, right-handed, unimpaired subjects participated in this study. Each subject performed 10 trials of the tracking task to become familiar with the task. When subjects became familiar with the task, they performed the tracking task with different PI gains. In order to determine the range of PI gains, a pilot study involving 3 subjects was performed. During the pilot study it was observed that PI gains which were above 0.0002 for P and 0.0007 for I resulted in jerky motion. Furthermore, the subjects could not track the desired motion trajectory well when the PI gains were selected below 0.00005 for Proportional (P) and 0.0001 for Integral (I). Thus, in order to generate training data for the ANN, the following 5 PI gains were used that covered the range of the upper and lower limits of the PI gains described above (Table 1). Note that all these 5 sets of gains could be used for each subject to complete the desired task. Each subject performed the task with these 5 PI gains.

Table 1. Control Gains

Control Gains	Proportional	Integral
PI_1	0.0002	0.0007
PI_2	0.00015	0.00055
PI_3	0.0001	0.0004
PI_4	0.000075	0.00025
PI_5	0.00005	0.0001

In order to determine the suitable control gains for each subject, the smoothness of interaction index (*SII*) is calculated as follows:

$$SII = \left| \frac{F_i(t) - F_i(t-1)}{\Delta t} \right| = \left| \frac{dF}{dt} \right| \tag{25}$$

where F_i is the force applied by the subject and $F_i(t) - F_i(t-1), \Delta(t)$ are the changes in force, and in time, respectively. In order to avoid the noise amplification in differentiation of the F_i data, we use an eighth-order butterworth low-pass filter. The rate of change of the force applied by the subject demonstrates the smoothness of the interaction between the robot and the subject. Then the mean of SII, $\overline{SII}$, was calculated to determine the average smoothness of interaction during the execution of the task using Equation (26). The less variation in subject applied force implied smooth interaction. Thus, the control gains which resulted in minimum $\overline{SII}$ was selected as the suitable control gains for that subject.

$$\overline{SII} = mean(SII) \tag{26}$$

Experiment 2: The purpose of this experiment is to evaluate the effectiveness of the proposed ANN based PI gain scheduling controller to result in smooth interaction.

Experiment 1 was used to demonstrate that there was a need to change control gains for each subject to result in smooth interaction. Here, we first demonstrate that even for the same subject changing control gains during different phases of the task may result in smooth interaction. Then, we show that the proposed ANN based PI gain scheduling controller could be used to determine and apply such control gains to result in smooth interaction for different subjects as well as for the same subject for different parts of the motion of the same task.

4. Results

The main focus of this paper is to present the new control framework which is shown in Fig. 3. However, the basic assistive controller (as shown in Fig. 2) is an important part of this new control framework that is responsible for generating the commanded force to complete

the task. Hence we first summarize the validation of the basic assistive controller from our previous work [17], [36] for the sake of completeness of the results. Then the results of the experiments described in Section 3.4 are provided.

4.1. Validation of the Basic Assistive Controller

The basic assistive controller has been previously verified in two ways. First, we asked the robot to apply a desired force on a spring-damper system (Fig. 5) [36]. We presented the force response of the controller in Fig. 6. As can be seen, the robot was able to track the desired force using the basic assistive controller, which is shown in Fig. 6.

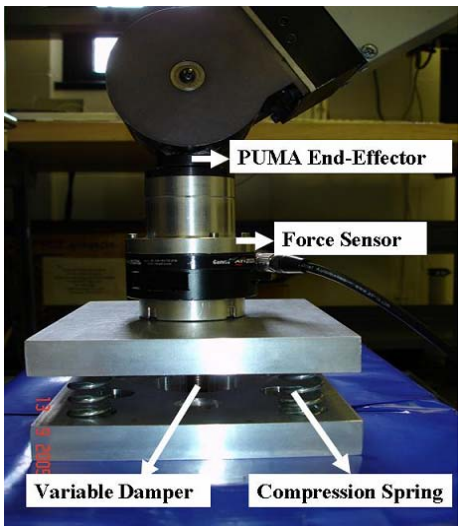


Fig. 5. Spring-damper system

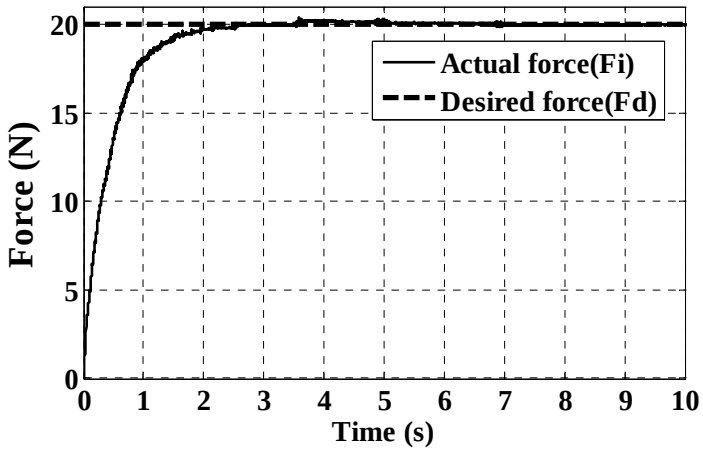


Fig. 6. Force Response

Second, after establishing the fact that the robot controller could indeed provide a desired force, we experimentally demonstrated that it could be used to generate appropriate

assistance to enable subjects to track a desired velocity [17]. Subjects were asked to track the desired motion displayed on the computer screen. If the subject could not track the desired motion in a defined velocity boundary, then the basic assistive controller was activated otherwise there was no need to provide robotic assistance to the subject to track the desired velocity. $\dot{x}$ was selected as *0.02m/s*. The $\dot{x}_u$ and $\dot{x}_l$ were selected as 25% more and less of $\dot{x}$, which were *0.025m/s* and *0.015m/s*, respectively. The average velocity of the subject $(\dot{x}_{ave})$, the upper $(\dot{x}_{uave})$ and lower $(\dot{x}_{lave})$ velocity boundaries were calculated using (27) at every 5 seconds. 5 seconds were sufficient to estimate the progress of the subject.

$$\dot{x}_{ave} = \frac{1}{\left(\frac{tf-ti}{ts}\right)} \sum_{t=ti}^{tf} (\dot{x}(t)) \quad , \dot{x}_{uave} = \frac{1}{\left(\frac{tf-ti}{ts}\right)} \sum_{t=ti}^{tf} (\dot{x}_u(t)) \; , \dot{x}_{lave} = \frac{1}{\left(\frac{tf-ti}{ts}\right)} \sum_{t=ti}^{tf} \left(\dot{x}_l(t)\right) \tag{27}$$

where tf, ti and ts are the final time, starting time and sampling time, respectively. The condition $\dot{x}_{lave} < \dot{x}_{ave} < \dot{x}_{uave}$ was checked to decide the activation of the basic assistive controller. It could be observed from Fig. 7, which presented one of the subjects data, that the subject's average velocity (stars), which was calculated every 5 seconds using (27), was out of range at *A, B* and *C* points (Fig. 7). The controller was activated for the next 5 seconds between *A-A', B-B'* and *C-C'*. It could be seen that the subject's velocity was brought inside the desired range at *A', B'* and *C'* points (Fig. 7), which verified that the controller could provide sufficient assistance to the subject to bring the velocities inside the desired range.

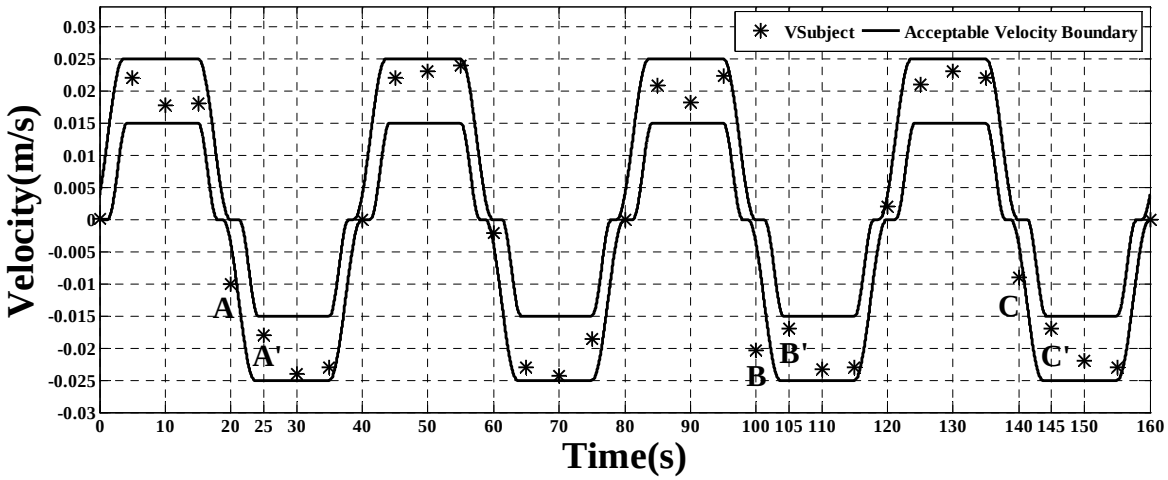


Fig. 7. Calculated Average Velocities

4.2. The ANN-based PI Gain Scheduling Controller – its need and usefulness

In Section 3.4 we describe two sets of experiments to demonstrate the benefit of the proposed controller. Here we present the results of those experiments. Note that following experiments were conducted using the minimum-jerk trajectory which is given in (24) .

Experiment 1: The purpose of this experiment is to demonstrate that each subject may require different control gains for the same task in order to result in smooth interaction.

In order to demonstrate different control gains resulted in different SII for each subject $\overline{SII}$ was calculated using (26). 10 subjects performed the position tracking task (described in 3.2) using the basic assistive controller (described in Section 2.1) with 5 different PI gains that were given in Table 1. The desired position trajectory was calculated using (24). In here, x_i, x_f and d was selected as *0, 0.3m* and *20* s, respectively when the subject was asked to move forward, and *0.3m* , *0 m* and *20* s, respectively when the subject was asked to move backward. Table 2 presented the $\overline{SII}$ calculated for each subject with 5 PI gains. It could be observed from Table 2 that different SII were obtained for different control gains for the same subject. Furthermore, same control gains resulted in different SII for different subjects. In order to test whether the difference between $\overline{SII}$ values were obtained by chance or not, we had used one-way ANOVA test. ANOVA test is used to determine whether the means of groups are statistically different from each other [41]. ANOVA results had been presented in Table 3a and 3b. Table 3a showed that the difference in $\overline{SII}$ for each subject was statistically significant within *99%* confidence level and was not obtained by chance. This implied that the $\overline{SII}$ difference was due to the change in control gains. Table 3b showed that the difference in $\overline{SII}$ for different subjects for the same set of control gains was statistically significant within *99%* confidence level, which implied that the change in $\overline{SII}$ was due to the difference in subjects. In other words, the above results imply that for each subject, $\overline{SII}$ could be reduced by selecting the control gains appropriately.

Table 2. $\overline{SII}$ Values for 10 Subjects with 5 Different PI Gains

Subject	PI_1	PI_2	PI_3	PI_4	PI_5
S1	3.92	4.03	4.24	5.05	5.36
S2	2.63	2.93	2.69	4.09	5.42
S3	2.08	2.66	2.19	4.03	2.89
S4	5.13	4.37	4.38	5.77	5.79
S5	2.17	2.19	2.44	2.50	2.33
S6	3.22	2.84	3.86	2.61	3.59
S7	6.03	4.27	5.84	5.05	4.52
S8	6.34	8.67	4.51	4.40	4.19
S9	4.05	4.10	4.91	5.36	3.11
S10	7.01	6.34	6.41	6.55	7.13

Table 3a. ANOVA Analysis for Each Subject with 5 Different PI Gains, **Table 3b.** ANOVA Analysis for one PI Gain Set with 10 Subjects

a

Subject	F	P-value	F critical
S1	913.78	p<.01	3.32
S2	3964.48	p<.01	3.32
S3	3851.18	p<.01	3.32
S4	882.22	p<.01	3.32
S5	375.38	p<.01	3.32
S6	3178.70	p<.01	3.32
S7	692.22	p<.01	3.32
S8	3583.91	p<.01	3.32
S9	1781.10	p<.01	3.32
S10	1653.13	p<.01	3.32

b

Control Gain	F	P-value	F critical
PI_1	6782.38	p<.01	2.41
PI_2	7742.45	p<.01	2.41
PI_3	4617.59	p<.01	2.41
PI_4	2957.39	p<.01	2.41
PI_5	3964.76	p<.01	2.41

Experiment 2: The purpose of this experiment is to evaluate the effectiveness of the proposed ANN based PI gain scheduling controller to result in smooth interaction

Experiment 1 results demonstrated that one needs to change the control gains that results in smooth interaction for each subject. It would be ideal if the control gains could be mapped directly to SII. However, as shown in Table 2, similar SII characteristics could also be obtained using different gains for different subjects. For example, the same $\overline{SII}$ was obtained for S3 with PI_3 and for S5 with PI_2. Thus, a mapping parameter was needed to differentiate among subjects. Human arm characteristics could be used as such mapping parameters. In this work, ANN was designed to map human arm stiffness as inputs and suitable control gains that result in smooth interaction as outputs. Thus, ANN required the knowledge of stiffness parameter, which was determined using the system identification method described in Section 2.2.

The accuracy of the system identification method had been demonstrated with a spring-damper system in our prior work [36]. A traditional PI based force controller was used to generate a constant contact force on different environments, where ARX model was used to

estimate environment parameters (stiffness and damping coefficients). Estimated stiffness values were very close (within 4.28% range) to the actual ones.

The force and position data were recorded to determine the stiffness of each subject using (10). The precise estimation of the stiffness was dependent on the movement of the equilibrium point. It was discussed in [27] that a small movement of the equilibrium point was desirable because it did not influence the accuracy of the value of the estimated stiffness for tasks that were performed with low velocity *(0.02m/s or less)*. Therefore, the duration of the equilibrium shifts to determine the equilibrium force and position (F_{eq} and x_{eq}) was selected as *0.01s* to keep the movement of the equilibrium point small.

It was noticed that the human arm stiffness varied during the task. While the exact variations were different for different subjects, the nature of the variation was similar. For every subject, the stiffness could be classified into four parts. In the first part, when the task began, the subject initiated movement towards the forward direction and the stiffness increased and reached a peak. Then the stiffness started decreasing and sometimes became negative as the motion progressed (Part II). When the subject started moving in backward direction, the stiffness started increasing and reached a peak (Part III). Finally the stiffness started decreasing as the motion progressed (Part IV) (Fig. 8). Negative stiffness could be due to when contracted muscles were stretched at high velocity as discussed in [42]-[43]. In the literature muscle stiffness in the ankle joint of humans and monkeys had been reported as negative when the muscle force rose higher during muscle shortening than muscle lengthening [42]-[43].

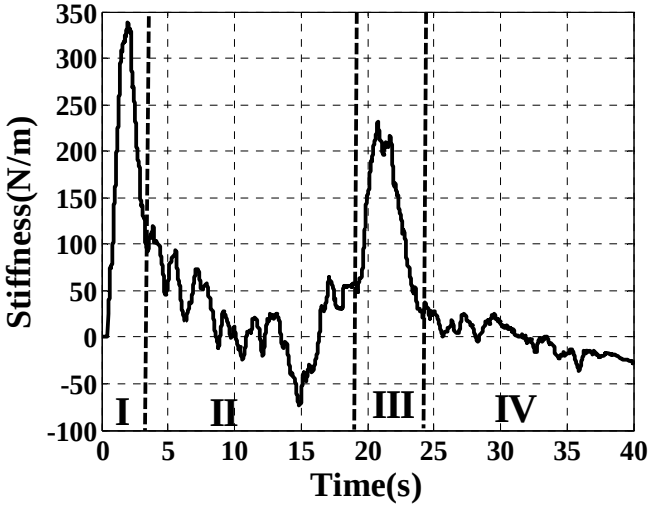


Fig. 8. Stiffness of Subject 1

The stiffness characteristics were similar for all 10 subjects as shown in Fig. 8. Note that the ANN was trained with stiffness as the input and the suitable control gains as the outputs. Since stiffness varied during the motion, the subject might need different control gains at different parts of the motion to obtain smooth interaction. Thus, the stiffness and the suitable control gains could be obtained separately for each part of the motion (e.g., Parts I, II, III and IV) and then used to train ANN. The control gains that resulted in minimum $\overline{SII}$ among 5 PI gains for each part of the motion were selected as the suitable control gains. Table 4 presented the suitable PI gains for each subject. As it could be seen from Table 4 that one subject might need different control gains during different phases of the task to obtain smooth interaction. Table 4 implied that the motion would result in higher $\overline{SII}$ if any one set of control gains were applied throughout the entire motion. For example, S4 would need PI_2 for Part I followed by PI_3 for Part II, followed by PI_2 for Part III and finally PI_4 for Part IV that would result in the minimum $\overline{SII}$ for the set of control gains.

Table 4. Suitable PI Gain for Each Part of the Motion for 10 Subjects

Subjects	I	II	III	IV
S1	PI_2	PI_1	PI_2	PI_1
S2	PI_2	PI_3	PI_2	PI_3
S3	PI_1	PI_1	PI_3	PI_2
S4	PI_2	PI_3	PI_2	PI_4
S5	PI_2	PI_2	PI_1	PI_1
S6	PI_3	PI_2	PI_3	PI_2
S7	PI_1	PI_1	PI_1	PI_4
S8	PI_1	PI_3	PI_1	PI_5
S9	PI_5	PI_5	PI_2	PI_5
S10	PI_3	PI_3	PI_1	PI_5

In order to represent subject's arm characteristics, we extracted three features of the stiffness K, which were the mean K_μ(Mean), the standard deviation K_σ (Std) and the maximum K_{max} (Max). We assumed that these 3 features were sufficient to capture the variation of the stiffness of the arm during the motion for the initial investigation. However, there could be additional features that might need to be explored in the future work. K_{max}, K_μ, and K_σ were calculated and presented in Table 5.

Table 5. Stiffness Values for Each Part of the Motion for 10 Subjects

	I			II			III			IV		
Subjects	**Max**	**Mean**	**Std**	**Max**	**Mean**	**Std**	**Max**	**Mean**	**Std**	**Max**	**Mean**	**Std**
S1	373.6	125.76	117.49	93.45	16.79	36.56	187.66	108.52	59.85	37.87	-1.65	17.10
S2	56.28	27.26	17.89	38.51	23.25	7.99	49.07	18.16	13.95	25.73	8.55	7.89
S3	102.37	44.09	31.86	20.63	-4.98	15.88	77.46	43.32	26.95	17.99	4.39	4.32
S4	149.41	78.22	40.69	45.69	-7.52	28.66	129.34	69.61	32.41	99.25	22.69	17.08
S5	89.56	36.72	37.69	45.054	18.30	11.14	70.52	10.07	40.67	35.82	4.73	15.10
S6	283.90	84.01	79.71	163.68	10.39	24.35	95.29	65.46	15.13	42.05	10.98	11.98
S7	195.19	75.40	58.55	42.29	15.97	11.30	167.37	78.93	57.6	168.21	39.71	34.39
S8	410.25	150.73	167.2	87.84	7.13	29	313	242.3	78.51	162.46	20.11	21.29
S9	321.77	105.32	106.66	66.34	29.83	15.50	88.05	13.75	26.47	88.05	25.21	18.70
S10	174.86	94.74	53.85	63.68	21.29	18	182.37	89	60	231.19	70.12	59.34

In order to demonstrate the evaluation of the proposed ANN-based PI gain scheduling controller first we trained the ANN and then used it in the real-time experiment to demonstrate that the ANN could predict control gains that result in smooth interaction for new subjects, whose data were not included in the training. 6 new subjects were invited to the laboratory, one female and five males, 26-30 years old, right-handed, unimpaired subjects. The ANN was trained offline using the stiffness as inputs (Table 5) and control gains as outputs (Table 4). The Levenberg-Marquardt (LM) algorithm (which was given as *trainlm* in Matlab Neural Network Toolbox) was used to train the ANN, which was able to obtain lower mean squared errors than any of the other algorithms available in the toolbox. In this application the error goal, which was the mean-squared error (MSE) [41], was selected as 10^{-8} for both the P and the I gains. To determine the appropriate number of hidden neurons, the LM backpropagation networks with 3, 5, 7, 10 and 15 hidden neurons were evaluated. The MSE between the outputs of the networks (P' and I' gains), which was trained with 3, 5, 7, 10 and 15 hidden neurons, and the target outputs (P and I gains – given in Table 4) were calculated. The MSE for the number of neurons 7 for both P and I gains was the closest to the 10^{-8} (Table 6), thus a 7 neuron hidden layer feedforward LM backpropagation ANN was selected for this application. We designed two such ANN – one for the P-gain prediction and one for the I-gain prediction in order to improve the accuracy of the ANN. The time for neural network (offline) training in Matlab using a 1.2GHz PC for the selected number of neurons was *0.82s* for P and *0.77s* for I.

Table 6. Mean-Squared Error (MSE) for LM Backpropagation Network

Number of Hidden Neurons	Goal*(10^{-8}) P	Goal*(10^{-8}) I
3	1.85	1.89
5	0.90	1.29
7	**0.30**	**0.50**
10	0.45	1.13
15	3.45	4.33

In order to verify whether the predicted gains would indeed result in smooth interaction during the task performance, 6 subjects were asked to perform the tracking task using ANN-based PI gain scheduling controller (shown in Fig. 3) in real-time. The initial gains were chosen to be the same for every subject. The controller was required to change the control gains based on the subject's arm stiffness to obtain smooth interaction. Every *0.5s* data was used to estimate K, which was then used to determine the K_{max}, K_{μ}, and K_{σ} at every sampling time using a sliding window. The control gains were predicted continuously, however it may not be required to switch control gains frequently because the stiffness of the subjects may not change rapidly. In this work, we choose a fixed switching time of 2 seconds for gain switching. However, the effect of switching time on smooth interaction will need to be explored in a systematic manner in the future work. In order to analyze whether the predicted gains resulted in smooth interaction, we compared the following $\overline{SII}$ values in Table 7. $\overline{SII}_1$ represented the minimum rate of change of the force applied by subject when the most appropriate control gains were selected manually from the chosen set of control gains for each subject. Any other control gains from that set would result in higher rate of change of the force applied for that subject. $\overline{SII}_2$ represented the highest of those $\overline{SII}$. It should be noted that we argue in this paper that even though there could be multiple sets of control gains that could be used to complete the task, specific sets of gains would be more suitable (in terms of smooth interaction) for specific subjects. The idea here was to verify whether the ANN predicted gains resulted in smooth interaction values better than $\overline{SII}_2$, and possibly closer to the $\overline{SII}_1$ values obtained through manually tuned best control gains. In this quest, we define $\overline{SII}\,midpoint = \dfrac{(\overline{SII}_1 + \overline{SII}_2)}{2}$ as the mid-point metric to determine whether the $\overline{SII}$ obtained were closer to $\overline{SII}_1$. In Table 7, $\overline{SII}_3$ was obtained when the PI gains were changed at every 2 seconds using the proposed ANN-based PI gain scheduling controller. It could be seen from Table 7 that using the proposed ANN-based PI gain scheduling controller,

it was possible to predict the control gains which resulted in smooth interaction that was better than $\overline{SII}_{midpoint}$ for all subjects. Furthermore, we had determined the median of the $\overline{SII}_{median}$ obtained with the 5 PI gains. The median is used to indicate the center point of a distribution. In here, median was used to represent the center $\overline{SII}$ value obtained among 5 different PI gains. Thus, if we could reduce the $\overline{SII}$ which resulted in less than the $\overline{SII}_{median}$, then we could say that the $\overline{SII}$ resulted with the predicted control gains were in the lower range of $\overline{SII}_{median}$. This would confirm that the $\overline{SII}_3$ obtained using ANN-based PI gain scheduling controller were closer to $\overline{SII}_1$ obtained through manually tuned best control gains. As it could be seen from Table 7 $\overline{SII}_3$ was less than $\overline{SII}_{median}$ for all subjects except S11 *(83.3% success)*. We had also determined how much the proposed controller was able to improve $\overline{SII}$ from the maximum one $\overline{SII}_2$. Equation (28) was used to determine the percentage $\overline{SII}$ improvement. It could be seen from Table 7 that the $\overline{SII}$ improvement was in the range of *24 %-33 %*.

$$SII\ Improvement = \left(\frac{\overline{SII}2 - \overline{SII}3}{\overline{SII}2} \right) * 100 \tag{28}$$

Table 7. Comparison of $\overline{SII}$ Values

Subjects	$\overline{SII}_1$	$\overline{SII}_2$	$\overline{SII}_{midpoint}$	$\overline{SII}_{median}$	$\overline{SII}_3$ (Switch at Every 2 s)	$\overline{SII}$ Improvement (%)
S11	3.62	6.88	5.25	4.16	4.62	32.86
S12	3.51	5.01	4.26	3.77	3.53	29.56
S13	3.00	4.61	3.81	3.87	3.50	24.02
S14	3.30	4.87	4.09	3.58	3.43	29.57
S15	4.70	6.96	5.83	5.34	4.81	30.87
S16	3.85	6.69	5.27	5.36	4.86	27.39

Finally, Figs. 9a and b presented PI gain switching at every 2 seconds for S3 as an example. Note that sometimes the gains stayed at the previous values when the ANN predicted the same gains.

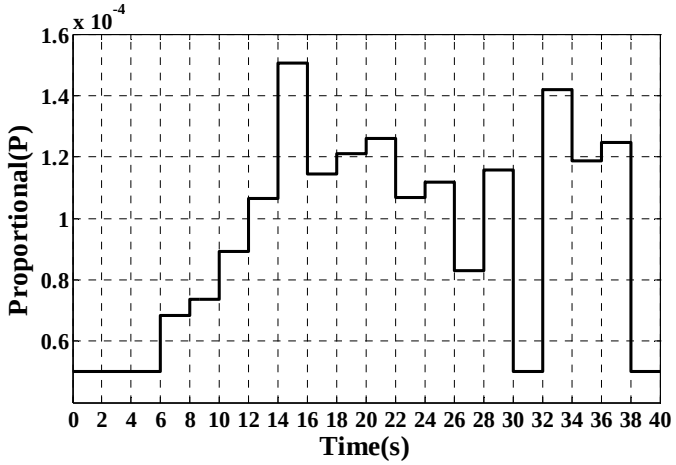


Fig. 9a. Proportional Gain (P) Changes for S3

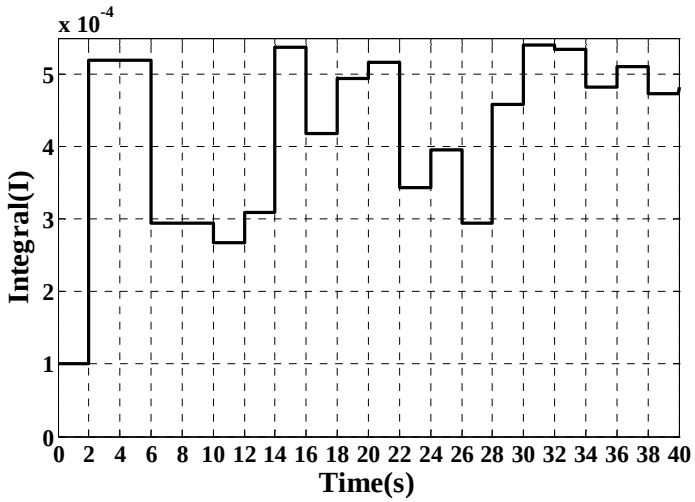


Fig. 9b. Integral Gain (I) Changes for S3

We had also noticed that the percentage error between the $\overline{SII}$ obtained from best manually tuned control gains, i.e., $\overline{SII}_1$, and $\overline{SII}_3$, which was obtained by the ANN-based gain scheduling controller, were very close for 3 subjects (less than *4%*, which were *0.45%, 3.79%*, and *2.37%* for S12, S14 and S15, respectively). In order to test whether the closeness of $\overline{SII}_1$, and $\overline{SII}_3$, were obtained by chance or not, we had used the t-test, which was used to assess the probability of getting close mean values by chance [41]. Alpha level was selected as *0.01*. It could be seen from Table 8 that the difference between two means was statistically significant within 99% confidence level so that they were not obtained by chance.

Table 8. t-test Evaluation of $\overline{SII}_1$ and $\overline{SII}_3$

Subjects	t-value	P(T<=t) two-tail	t Critical two-tail
S12	5.19	2.83E-09	2.58
S13	23.22	1.2E-118	2.58
S15	15.99	2.16E-09	2.58

5. Discussion and Conclusion

In this work, we design a control framework that will automatically adjust the control gains for each subject such that the resultant interaction dynamics between the subject and the robot could result in smooth interaction. In order to demonstrate the smooth interaction, the rate of change of the force applied by the subject is evaluated. The results in Section 4 show that control gains could play an important role to obtain the smooth interaction. For example, for the same motion, different control gains result in different rate of change of force applied by the subject for the same subjects. In addition, for the same motion and for the same control gains, different rate of change of subject applied force are obtained for different subjects. Thus we design a new automated method of determining suitable control gains for each subject that will result in smooth interaction. This method could relieve the burden on a therapist to manually tune the control gains for each subject.

In this paper an ANN-based PI gain scheduling controller is presented which was used to provide robotic assistance to the subjects in such a manner that the resulting interaction between the robot and the patient is smooth. The controller improves the robotic assistance by automatically adjusting the control gains for each subject which results in smooth interaction. We first demonstrate that different control gains are needed for each subject in order to perform the task in a smooth manner. Then, the proposed ANN-based PI gain scheduling controller is evaluated with experiments conducted on unimpaired subjects to verify whether it is possible to automatically determine the control gains for each subject that results in smooth interaction. The rate of change of the force applied by the subject that is obtained using the ANN-based PI gain scheduling controller is i) always less than the rate of change of the force applied by the subject that is obtained when manual tuning of the control gains is improper ($\overline{SII}_2$ in Table 7); ii) always less than the $\overline{SII}_{midpoint}$ (i.e., success rate *100%*) that indicates that the automated controller results in smooth interaction that are closer to the rate of change of the force applied by the subject that is obtained from best manually tuned control gains; and iii) less than $\overline{SII}_{median}$ for 5 out 6 subjects (i.e., success rate *83%*) that indicates that the automated controller could produce smooth interaction that are in the

lower half of the smooth motion. It has also been shown that the proposed controller was able to improve the smoothness of interaction from to the maximum one $\overline{SII}_2$ in the range of *24%-33 %*.

While the proposed controller shows promise in generating robotic assistance in such manner that the interaction between the subject and the robot is smooth, there are several issues that need to be explored in the future to improve the performance of the controller. In the current work, we use only stiffness as the human arm parameter to train the ANN. It is possible that mass and damping coefficient may be needed to better capture the arm characteristics of the stroke patients. In addition, the number and the nature of the features (e.g., here we used 3 features of the stiffness: the mean K_μ, the standard deviation K_σ and the maximum K_{max}) to train the ANN could be explored to investigate the improvement of the gain prediction. The presented results are based on the training of the ANN with the data of 10 subjects. With the increase in training data set the prediction accuracy of the ANN will likely to be increased. As the robot interacts with more number of subjects, the training data sets will be increased. The time for neural network (offline) training is less than *1* second - *0.82* seconds for the P gain and *0.77* seconds for the I gain in Matlab using a 1.2GHz PC. Once offline training is completed the ANN is used online in conjunction with the human arm parameter estimation module. Another issue that may improve the performance of the controller is to determine an optimal gain switching strategy. In this work, we use a fixed switching time for the control gains. However, it is possible to design a gain switching sequence that switches the predicted gains when a predefined threshold is exceeded.

The presented controller could provide continuous robotic assistance in the early phases of the rehabilitation therapy that could motivate patients to use their impaired arm. Especially for low-functioning patient who cannot initiate arm motion, such assistance could help them participate in therapies that require extensive use of their impaired arms. For example, the Constraint-Induced Movement Therapy (CIMT) [3]-[4] restrains the less affected arm of the patient using a sling or a glove and forces them to use their impaired arm to practice shaping.

Note that the presented controller is not specific to the proposed rehabilitation robotic system but can be integrated with other previously proposed rehabilitation robotic systems. We are aware that a PUMA 560 robot might not be ideal for rehabilitation applications. However the measures we have taken (Section 3) to ensure safety may allow the system to be safe and useful for the patients.

Acknowledgments

We gratefully acknowledge the help of Dr. Thomas E. Groomes who is the Medical Director of Spinal Cord and Traumatic Brain Injury Program, and therapist Sheila Davy of Vanderbilt University's Stallworth Rehabilitation Hospital for their feedback about experimental setup and task design during this work. The work was supported by Vanderbilt University Discovery grant.

References

[1] Matchar, D.B. & Duncan, P.W. 1994. Cost of stroke, Stroke Clin Updates, Vol. 5, pp. 9-12.

[2] American Heart Association: Heart and Stroke Statistical Update, http://www.Americanheart.org/statistics/stroke.htm, 2006.

[3] Taub, E., Crago, J. E., & Uswatte, G. 1998. Constraint-induced movement therapy: A new approach to treatment in physical rehabilitation. Rehabilitation Psychology, Vol. 43, pp. 152-170.

[4] Taub, E.; Uswatte, G. & Pidikiti, R. 1999. Constraint-Induced Movement Therapy: A New family of techniques with broad application to physical rehabilitation – a clinical review. J. of Rehab. Research and Development. Vol. 36, pp. 237-251.

[5] Lum, P.S.; Burgar, C.G.; Van der Loos, H.F.M.; Shor, P.C.; Majmundar M.; Yap R.2006. MIME robotic device for upper-limb neurorehabilitation in subacute stroke subjects: A follow-up study . J. of Rehab. Research & Development, 43(5):631-642.

[6] Johnson M.J., Wisneski K.J., Anderson J., Nathan D., Smith R. 2006. Development of ADLER: The Activities of Daily Living Exercise Robot. First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob 2006): 254 - 259.

[7] Loureiro, R.C.V., Johnson, M.J., Harwin, W.S. 2006. Collaborative Tele-rehabilitation: A Strategy for Increasing Engagement. First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob 2006): 859 – 864.

[8] Kahn, L.E.; Zygman, M.L; Rymer, W.Z. & Reinkensmeyer, D.J. 2006. Robot-assisted reaching exercise promotes arm movement recovery in chronic hemiparetic stroke: a randomized controlled pilot study. Journal of NeuroEngineering and Rehabilitation, 3(12):1-13.

[9] Krebs, H.I.; Palazzolo, J.J.; Dipietro, L.; Ferraro, M.; Krol J.; Rannekleiv, K.; Volpe, B.T. & Hogan N. 2003. Rehabilitation Robotics: Performance-Based Progressive Robot-Assisted Therapy. Autonomous Robots, 15(1):7-20.

[10] Krebs, H. I.; Ferraro, M., Buerger, S.P., Newbery, M. J., Makiyama, A., Sandmann, M.; Lynch, D.; Volpe, B. T. & Hogan, N. 2004. Rehabilitation robotics: pilot trial of a

spatial extension for MIT-Manus. Journal of NeuroEngineering and Rehabilitation, 1(5):1-15.

[11] Loureiro, R.; Amirabdollahian, F.; Topping, M.; Driessen, B. & Harwin, W. 2003. Upper limb mediated stroke therapy - GENTLE/s approach. Autonomous Robots. Vol. 15, pp. 35-51.

[12] Kahn L.E; Lum P.S.; Rymer W.Z; Reinkensmeyer D.J. 2006. Robot-assisted movement training for the stroke-impaired arm: Does it matter what the robot does? J. of Rehab. Research & Development, 43(5):619-630.

[13] Flash T., Hogan N. 1985. The coordination of arm movements: an experimentally confirmed mathematical model. J Neurosci, Vol. 5, pp. 1688—1703.

[14] Hogan N. 1988. Planning and execution of multijoint movements, Can J Physiol Pharmacol, Vol. 66, pp. 508-517.

[15] Rohrer B. ,Fasoli S. , Krebs H.I. ,Hughes R. ,Volpe, Frontera W.R. , Stein J., Hogan N. 2002. Movement Smoothness Changes during Stroke Recovery. The Journal of Neuroscience, 22(18):8297–8304.

[16] Stein R.B., Cody F.W., Capaday C. 1988. The trajectory of human wrist movement. J Neurophysiol, 59:1814-1830.

[17] Erol D. and Sarkar N. (accepted to be published in vol. 4, no.3, September 2007). Design and Implementation of an Assistive Controller for Rehabilitation Robotic Systems .International Journal of Advanced Robotic Systems.

[18] Culmer P, Jackson A, Richardson R, Bhakta B, Levesley M, Cozens A. 2005. An admittance control scheme for a robotic upper-limb stroke rehabilitation system. International Conference on Engineering in Medicine and Biology society (EMBS), pp. 5081 - 5084.

[19] Sciavicco, L. & Siciliano, B. 1996. Modeling and Control of Robot Manipulators, McGrawHill.

[20] Dolan J.M., Friedman M.B., Nagurka M.L. 1993. Dynamic and loaded impedance components in the maintenance of human arm posture. IEEE Trans Syst Man Cybern, 23(3):698–709.

[21] Flash T., Mussa-Ivaldi F. 1990. Human arm stiffness characteristics during the maintenance of posture. Exp Brain Res, 82(2):315–326.

[22] Gomi H., Koike Y., Kawato M. 1992. Human Hand Stiffness during Discrete Point-to-point Multi-joint Movement. IEEE Engineering in Medicine and Biology Society, Vol.14, pp. 1628-1629.

[23] Mussa-Ivaldi F., Hogan N., Bizzi E. 1985. Neural, mechanical, and geometric factors subserving arm posture in humans. J Neurosci, 5(10): 2732–2743.

[24] Speich J. E., Shao L., Goldfarb M. 2005. Modeling the human hand as it interacts with a telemanipulation system. Mechatronics. 15(9):1127-1142.

[25] Rahman M.M., Ikeura R., Mizutani K. 2000. Control characteristics of two humans in cooperative task. IEEE Int. Conf. on Sys. Man and Cybernetics, Vol: 2, pp. 1301-1306.

[26] Tsuji T., Goto K., Moritani M., Kaneko M., Morasso P. 1994. Spatial Characteristics of Human Hand Impedance in Multi-Joint Arm Movements. IEEE International Symposium on Intelligent Robots and Systems, pp. 423-430.

[27] Tsumugiwa T., Yokogawa R., Hara K. 2002. Variable Impedance Control Based on Estimation of Human Arm Stiffness for Human-Robot Cooperative Calligraphic Task. IEEE International Conference on Robotic and Automation, pp. 644-650.

[28] Gribble P.L., Ostry D.J., Sanguineti V., Laboissière R. 1998. Are Complex Control Signals Required for Human Arm Movement?. Journal of Neurophysiology, Vol. 79: pp. 1409-1424.

[29] Mukaibo Y., Park S. and Maeno T. 2004. Equilibrium Point Control of a Robot Arm with a Double Actuator Joint. International Simposium on Robotics and Automation.

[30] Shadmehr R. 1998. The Equilibrium Point Hypothesis for Control of Movement. Dept. of Biomedical Engineering Johns Hopkins University.

[31] Bizzi E., Accordnero N., Chapple W., Hogan N. 1984. Posture control and trajectory formation during arm movement. J. Neurosci, Vol. 4, pp. 2738– 2744.

[32] Flanagan J. R., Ostry D. J., Feldman A. G. 1993. Control of trajectory modifications in target-directed reaching. J. Mot. Behav, Vol. 25, pp. 140–152.

[33] Ljung L. 1999. System Identification Theory for the User, Prentice Hall, 2nd Ed.

[34] Wang J., He Q.P., Qin S.J., Bode C.A., Purdy M.A. 2005. Recursive Least Squares Estimation for Run-to-Run Control with Metrology Delay and Its Application to STI Etch Process. IEEE Transactions on Semiconductor Manufacturing,18(2):309-319.

[35] Graebe S.F., Ahlèn A. 1995. Bumpless Transfer in W. S. Levine (Ed.). The Control Handbook, pp. 381-388. CRC.

[36] Mallapragada V., Erol D., Sarkar N. 2006. A New Method for Force Control for Unknown Environments. IEEE/RSJ International Conference on Intelligent Robots and Systems.

[37] PUMA 560 Related Sites on the Internet, http://www.ee.ualberta.ca/~jasmith/puma/pumasites.html.

[38] ATI Industrial Automation, http://www.ati-ia.com/.

[39] Carey J.R., Bhatt E., Nagpal A. 2005. Neuroplasticity Promoted by Task Complexity, Exercise and Sport Science Review, Vol. 33, pp. 24-31.

[40] Carey J., Durfee W., Bhatt E., Nagpal A., Weinstein S., Anderson K. , Lewis S. 2006. Tracking vs. movement Telerehabilitation training to change hand function and brain reorganization in stroke, Submitted to Neurorehabilitation and Neural Repair.

[41] Milton S. J., Arnold J. C.2003. Introduction to Probability and Statistics: Principles and Applications for Engineering and the Computing Sciences, McGraw Hill, 4th Edition.

[42] Rack P. M. H. 1981. Neural control of muscle length. In Handbook of Physiology, vol.2, section.1, Motor Control, part 1, ed. Brookhart, j. m. & Mountcastle, V. B., Physiological Society, Bethesda, MD, USA, pp. 229-256.

[43] Dyhre-Poulsen P, Simonsen E. B. and Voigt M. 1991. Dynamic control of muscle stiffness and H reflex modulation during hopping and jumping in man. J Physiol, Vol. 437, pp. 287–304.

CHAPTER V: MANUSCRIPT 4

COORDINATED ROBOTIC ASSISTANCE FOR ACTIVITIES OF DAILY LIVING

Duygun Erol & Nilanjan Sarkar

(Submitted in Proceedings of International Conference on Intelligent Robots and Systems (IROS 2007))

Abstract

Recent research in rehabilitation indicates that the activities of daily living (ADL) focused tasks have shown significant increase in the motor recovery after stroke. This paper presents a new control approach to robot assisted rehabilitation for stroke patients that enables them to perform ADL. The control architecture is represented in terms of a hybrid system model combining a high-level controller for decision-making and two low-level assistive controllers (arm and hand controllers) for providing arm and hand motion assistance. Experimental results are presented to demonstrate the efficacy of the proposed control architecture.

Keywords: robot-assisted rehabilitation for activities of daily living tasks, coordination of arm and hand assistive devices, hybrid system model.

1. Introduction

Stroke is a highly prevalent condition [1], especially among the elderly that results in high costs to the individual and society [2]. According to the American Heart Association (2006), in the U.S., approximately 700,000 people suffer a first or recurrent stroke each year [1]. It is a leading cause of disability, commonly involving deficits of motor function. Recent literature supports the idea of using intense and task oriented stroke rehabilitation [3] which assumes that control of movement is organized around goal-directed functional tasks. Task-oriented approach demonstrated promising results in producing a large transfer of increased limb use to the activities of daily living (ADL) [4],[5]. One of the successful examples of task-oriented therapies is Constraint-Induced Movement Therapy (CIMT) [6], which has shown to reduce learned non-use and increase ADL functioning for stroke patients. The availability of such training techniques, however, is limited by the amount of costly therapist's time they involve and the ability of the therapist to provide controlled, quantifiable

and repeatable assistance to complex arm movement. Consequently, robot assisted rehabilitation that can quantitatively monitor and adapt to patient progress, and ensure consistency during rehabilitation may provide a solution to these problems.

Robotic devices are general-purpose aids used to assist, enhance and quantify rehabilitation therapy, which has been an active research area for the last few years. The robotic assistive devices used for arm rehabilitation are the MIT Manus [7], Mirror Image Movement Enabler (MIME) [8], Assisted Rehabilitation and Measurement (ARM) Guide [9] and GENTLE/s [10]. New rehabilitation therapy environments are under development to permit training of real-life functional tasks involving reaching, grasping [11]. Rutgers Master II-ND [12], a hand exoskeleton [13], the CyberGrasp [14], a pneumatically controlled glove [15] and a robotic device (HWARD) [16] are also being developed to facilitate the hand movement of stroke patients.

Even though existing arm and hand rehabilitation systems have shown promise of clinical utility, they are limited by their inability to simultaneously assist both arm and hand movements. This limitation is critical because in general, rehabilitation involving task-oriented training requires movement of the whole arm of the patient including hand movement, especially since the great majority of ADL carried out by the upper extremity requires participation of the hand and fingers. Robots that cannot simultaneously assist both arm and hand movements are of limited value in the task-oriented approach. In order to achieve the desired coordinated motion between arm and hand, we propose a control framework that consists of a high-level controller and two low-level device controllers (e.g., arm and hand controllers). Note that the presented control architecture is not specific to a given arm and hand assistive devices but can be integrated with any previously proposed assistive systems.

This paper is organized as follows. It first presents the overall control architecture in Section II. Then the rehabilitation robotic system is presented in Section III. The low-level assistive controllers and the high-level controller of the overall control architecture have been described in Section IV and Section V, respectively. Results of the experiment are presented in Section VI to demonstrate the efficacy of the control architecture. Section VII discusses potential contributions of this work and presents the conclusion.

2. Control Architecture

Let us first present the proposed control architecture in the context of generic ADL tasks that require coordination of both arm and hand movement (e.g., eating, drinking, combing, dressing etc.). However, the stroke patients may not be able to complete the ADL tasks because of their motor impairments in their arms and hands. Thus, low-level assistive controllers could be used to provide assistance to subjects' arm and hand movement and a high-level controller could be used for the coordination of these low-level assistive controllers. The nature of assistance given to the patients and coordination of the assistive devices, however, could be impacted by various events during the ADL task. Some examples of event could be: i) if the patient has reached the object, then the coordination must initiate grasping activity, ii) if the patient feels uncomfortable, then the execution of the task should be paused till the patient feels better, iii) if the assistive devices develop any fault, then the control actions needs to be adjusted for the safety of the patients. These set of information may require some adjustments of the planning of the task. As a result, both the arm and hand low-level assistive controllers need to be aware of these adjustments of the task to accomplish the therapy requirements. The high-level controller could be used to allocate task responsibility between the arm low-level assistive controller and the hand low-level assistive controller based on the task requirements and specific events that may arise during the task performance. The high-level controller in here plays the role of a human supervisor (therapist) who would otherwise monitor the task, assess whether the task needs to be updated and determine the activation of the assistive devices. However, in general, the high-level controller and the low-level assistive controllers cannot communicate directly because each may require different types of inputs and outputs. For example, a high-level controller may operate in the discrete domain whereas low-level assistive controllers may operate in the continuous domain. Thus an interface is required which can convert continuous-time signals to sequences of discrete values and vice versa. Hybrid system theory provides mathematical tools that can accommodate both continuous and discrete system in a unified manner. As a result, in this work, we take the advantage of using a hybrid system model to design our control architecture. A hybrid system model has three parts, a "Plant", a "Controller" (supervisor) and an Interface [17]. In order to avoid confusion about terminology, we call the "Controller" in hybrid system model a high-level controller in this paper. The continuous part, identified as the "Plant" represents both the arm low-level assistive controller and the hand low-level assistive controller. The Interface consists of a generator and an actuator,

which functions as analog-to-digital/digital-to-analog (AD/DA) adaptor, respectively. The interface accepts symbolic inputs via the actuator (control symbols) and produces symbolic outputs (plant symbols) via the generator. Note that actuator is not representing the actuator of the assistive devices.

In this paper, we propose a new control architecture that exploits hybrid system modeling techniques to provide robotic assistance for ADL tasks (Fig. 1). Hybrid control framework has been effectively used in other fields, such as industrial robotics, medicine, and manufacturing [18], however, it is applicability for rehabilitation purposes is new. We argue that the proposed control architecture based on hybrid system framework could be useful in coordinating the assistive device controllers in a safe and complex manner to satisfy a variety of ADL task requirements. As discussed in Section I that there is no existing controller that can coordinate multiple assistive devices needed for ADL tasks. The proposed controller is expected to address this need in the field of rehabilitation robotics.

In this architecture (Fig. 1), the sensory information from the arm assistive device (robot), the hand assistive device and the feedback from the human are monitored by the process-monitoring module through the interface to trigger the relevant plant events. Each plant event is represented as a plant symbol so that the high-level controller can recognize the event. Once the high-level controller receives the plant event through a plant symbol, the decision making module of the high-level controller generates sequences of control actions using its decision rules. The high-level controller is designed considering the requirements of ADL tasks and it can be easily modified and extended for new ADL task requirements. The decision of the high-level controller is sent to the low-level assistive controllers through the interface using the control symbols. Interface converts the control symbols to the plant inputs which are used to activate/deactivate the low-level assistive device controllers to complete the ADL task. This cycle continues to complete the therapy. The proposed control architecture is flexible and extendible in the sense that new events can be included and detected by simply monitoring the additional sensory information from the human, the arm device (robot), the hand device and accommodated by introducing new decision rules.

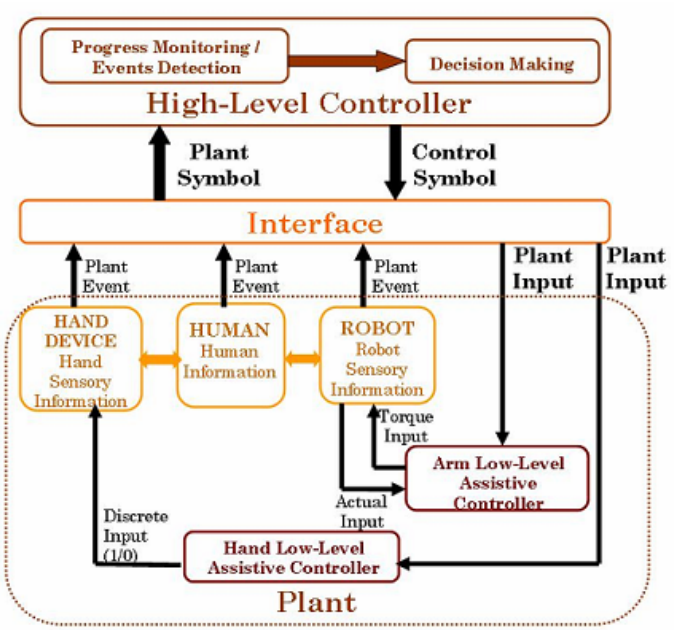


Figure 1 Control Architecture

3. Rehabilitation Robotic System

In order to implement and verify the proposed control architecture, we develop a rehabilitation robotic system that consists of a device for assisting arm movement and a device for assisting hand movement (Fig. 2A). The primary focus of this paper is to demonstrate how these two devices can be coordinated using a high-level controller for a given ADL task. In what follows, we first describe the arm assistive device, the hand assistive device and their sensory capabilities in this section.

3.1. Arm Assistive Device

A PUMA 560 robotic manipulator is used as the main hardware platform to provide assistance to the upper arm movement. We design a hand attachment device where the subject's arm is strapped into a splint and the robot is attached to that splint to provide assistance to the upper arm movement using the arm low-level assistive controller. Forearm padded aluminum splint (from MooreMedical), which ensures the subject's comfort, is used as a splint. We further design a steel plate with proper grooves that hold two small flat-faced electromagnets (from Magnetool Inc.) that are screwed on it. An automatic release (AU) rectifier controller (Magnetool Inc.) has been used to provide a quick release of these electromagnets. Ensuring safety of the subject is an important issue in designing a rehabilitation robotic system. Thus, in case of emergency situations, therapist/patient can press push button in order to demagnetize the magnets and to quickly release the patient's

arm from the robot to deal with any physical safety related events. More details of the arm assistive device could be found in [19].

3.2. Hand Assistive Device

A Power Grip Assisted Grasp Wrist-Hand Orthosis (from Broadened Horizons) is used as an assistive hand device to provide a functional key-pinch grasp by moving the index and middle fingers against a stabilized thumb, closing the grasp (Fig. 2Bi). This device is used to provide the ability to pick up, hold, and manipulate objects for a variety of most common ADL tasks. A purpose-built linear actuator attached to a JAECO-style durable aluminum splint at the metacarpophalangeal (MP) knuckle and wrist joint provides the force to open and close the fingers against the thumb. The splint is typically custom-built to the end users hand size measurements but can be easily adjusted to achieve an optimal fit. The linear actuator contains a micro-motor and matched set of planetary gears to slow the motor down to a functional speed and torque. The speed and force applied to open/close hand are variable and can be adjusted using the adjustable power supply (Fig. 2Bii) or using the computer software. Two 5V discrete logic inputs, splint connection, power and twin manual pushbuttons are connected to an orthotic controller (Fig. 2Biii). Two 5V discrete logic inputs act as an interface with the Matlab for the hand device. When the logic input is triggered within the software the orthosis and correspondingly the hand is opened or closed. Additionally, twin manual adaptive pushbuttons are used to provide the user the ability to manually control hand opening (extension) and hand closing (flexion) (Fig. 2Biv).

3.3. Contact Detection System

The interface forces between the hand device and the object are explicitly measured using the force-sensitive resistors (FSR) (Interlink Electronics, Inc.), which are placed on the fingertip (one on the thumb and the other one is on the intersection of index and middle finger as shown in Fig. 2C) to estimate the forces applied on the object during the grasping task. FSR 402 is selected for our application because its size is convenient for mounting on a fingertip. The pressure applied normal to the surface of an FSR can be measured as a change in the voltage across the FSR resistance which are recorded using National Instruments PCI-6031E data acquisition card with a sampling time of 0.001 seconds.

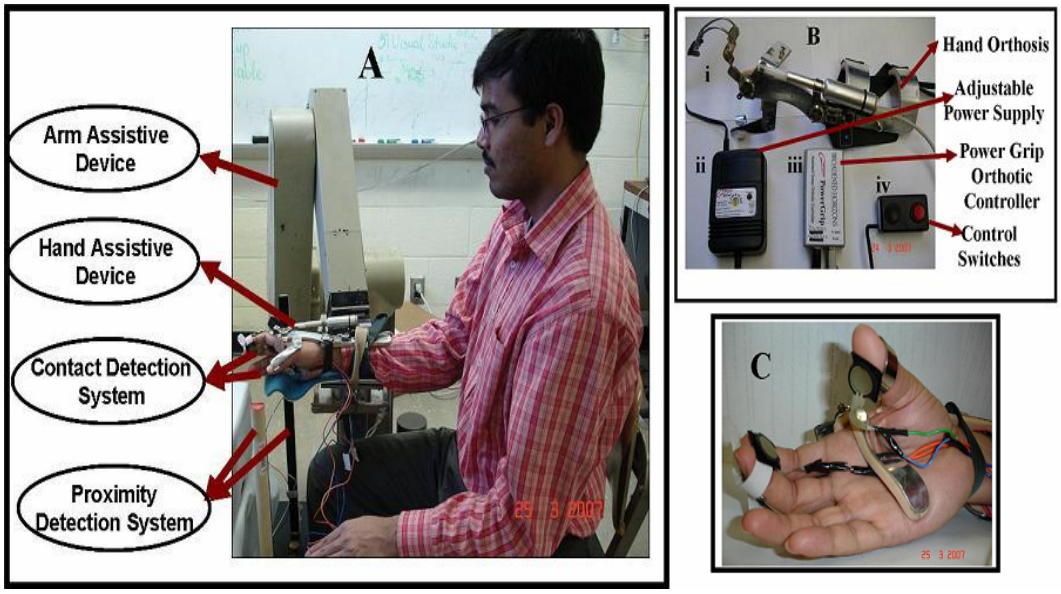


Figure 2A Rehabilitation Robotic System, **Figure 2B** Hand Assistive Device, **Figure 2C** Contact Detection System

3.4. Proximity Detection System

A proximity detection system (PDS) is designed in order to detect the closeness of the hand relative to the object to be grasped. PDS could be useful in providing the information to the high-level controller for planning of coordination decisions. PDS contains a phototransistor (sensitive to infrared light) and an infrared emitter (RadioShack, Inc). When the hand is between the emitter and the phototransistor, then the voltage, which is recorded using National Instruments PCI-6031E data acquisition card with a sampling time of 0.001 seconds, is increased.

4. Low-Level Assistive Controllers

4.1. Arm Low-Level Assistive Controller

The existing robotic rehabilitation systems for arm rehabilitation operate in either in robot task-space [7]-[10] or joint-space [20] to provide robotic assistance to the patients to follow a desired motion trajectory to complete a rehabilitation task. In this work, a proportional-integral-derivative (PID) position control is used as an arm low-level assistive controller for providing robotic assistance to a subject to complete the arm movement. The subject receives visual feedback of both his/her actual position and the desired position trajectories on a computer screen, which is placed in front of them. He/she is asked to pay attention to tracking the desired motion trajectory as accurately as possible. If the subject deviates from the desired motion, then arm low-level assistive controller provides robotic assistance to complement the subject's effort to complete the task as desired.

4.2. Hand Low-Level Assistive Controller

The hand assistive device is controlled in two ways, i) manually controlled by pressing control switches (Fig. 2Biv), ii) computer controlled by providing discrete values (1/0) to the discrete logic input. Subjects could hold on pressing red button to open their hand and black button to close their hand (Fig. 2Biv). On the other hand, discrete value 1 could be given to the digital output block in Matlab to activate hand device to open or close from the computer. The hand device opens or closes the hand with the specified power in the adjustable power supply (Fig. 2Bii) or from the computer. The subject is asked to follow the hand device speed as much as possible. If the subject could not follow the hand device movement, then the hand low-level assistive controller provides assistance to complement the subject's effort to open/close his/her hand. In here, the hand assistive device is controlled by the computer controlled way.

5. High-Level Controller

The high-level controller is designed to coordinate the arm and the hand devices in a safe manner. The decisions of the high-level controller are executed by the low-level assistive controllers to provide assistance to the subjects to complete the ADL task. Although the design methodology of the high-level controller presented here is general for a large class of ADL tasks, we choose one of these ADL tasks (drinking from a bottle (DFB)) to explain the design details of the high-level controller.

5.1. Task Description

We choose DFB task as an example to illustrate the proposed high-level controller. This task is chosen because it requires complex coordination of both arm and hand movement, which is common to many ADL tasks. This DFB task consists of several movements such as reaching towards the bottle, grasping the bottle, lifting the bottle from the table, drinking from the bottle, and placing the bottle back on the table movements [21]. We decompose DFB task into phases: i) reach towards the bottle while opening the hand, ii) reach the bottle, iii) close the hand to grasp the bottle, iv) move the bottle towards the mouth to drink using a straw, v) place the bottle back on the table, vi) open the hand to leave the bottle and vii) go back to starting position. At the beginning of the task the subject is asked to move towards the bottle with the help of the arm assistive device while he/she is opening his/her hand with the help of hand assistive device. The PDS, which has been described in details in section III-

A is placed in the two-thirds way of the reaching movement because it has been previously shown that the grasp component reached its maximum aperture approximately two-thirds (60%-80%) of the way through the duration of the reaching movement [22]. The placement of the detection system (60%-80% of the reaching movement) could easily be changed based on the patient's movement ability and the therapist choice. For example, low-level functioning patients could not be able to reach the maximum aperture by two-thirds of the reaching movement because of their movement inability, and thus PDS could be placed closer to the target object which provides additional time for the patients to reach their maximum aperture. When the subject's hand is detected by the PDS then the subject is asked to move towards the bottle to reach the bottle with the help of the arm assistive device. Later the subject is asked to grasp the bottle by closing his/her hand and the voltage value across the FSRs (Fig. 2C) is calculated to decide if the grasping is enough to hold the bottle. When the subject grasps the bottle, then he/she is asked to move the bottle to his/her mouth and drink water using a straw and then place the bottle on the table with the help of the arm assistive device. The subject is then required to open his/her hand to leave the bottle on the table with the help of hand assistive device. Finally, the subject is asked to go back to the starting position with the help of the arm assistive device. The subject performs the DFB few times to get familiar with the task. Note that similar task decomposition could be defined for any other ADL tasks.

5.2. Design Details of a High-Level Controller

The hybrid control systems consist of a plant and a discrete event controller (DES) connected to the plant via an interface. In hybrid control system, if the plant is taken together with the interface, then it is called DES plant model [17]. The DES controller, which is the high-level controller in this application, controls the DES plant. Let us first introduce the DES plant details and then describe the DES controller.

The DES plant model is a nondeterministic finite automaton, which is represented mathematically by $G = (\tilde{P}, \tilde{X}, \tilde{R}, \psi, \lambda)$. Here $\tilde{P}$ is the set of discrete states; $\tilde{X}$ is the set of plant symbols generated based on the event; and $\tilde{R}$ is the set of control symbols. $\psi : \tilde{P} \times \tilde{R} \to 2^{\tilde{P}}$ is the state transition function. For a given DES plant state and a given control symbol, the state transition function is defined as the mapping from $\tilde{P} \times \tilde{R}$ to the power set $2^{\tilde{P}}$, since for a given plant state and a control symbol the next state is not

uniquely defined. The output function, $\lambda : \tilde{P} \times \tilde{P} \to 2^{\tilde{X}}$, maps the previous and current states to a set of plant symbols. Plant state, $\tilde{P}$, is based on the set of hypersurfaces realized in the interface, which are not unique and decided considering the need of the therapy, and the capabilities of the rehabilitation robotic systems. Generally the available sensory information from the robotic systems and the input from the therapist and the subject provide the core of the set of the hypersurfaces which are given as:

$h_1(x) = voltage\_ir > 0,$

$h_2(x) = |x| \geq |x_{target}|$

$h_3(x) = (voltage\_fsr < voltage\_threshold) \wedge (handclose\_button == 0)$

$h_4(x) = (|x| \leq |x_{target}|) \wedge (handopen\_button == 0)$

$h_5(x) = (t == t_{hand})$

$h_6(x) = \theta_{lower} < \theta < \theta_{upper}$

$h_7(x) = \tau_{robot} \geq \tau_{rthreshold}$

$h_8(x) = \tau_{hand} \geq \tau_{hthreshold}$

$h_9(x) = (emergency\_button == 1)$

$h_{10}(x) = (pause\_button == 1)$

$h_{11}(x) = (pause\_button == 0) \wedge (emergency\_button == 0)$

where $voltage\_ir$ is the voltage recorded in the PDS system. x and x_{target} are the robot's and object's actual positions, respectively. $voltage\_fsr$ is the voltage across the FSRs. If the voltage across the FSRs drops below $voltage\_threshold$ then the subject should stop closing his/her hand. The values of $voltage\_threshold$ could be changed based on the grasp capabilities of the subject and the object characteristics. Manual hand open/close buttons could also be used by the subject's unimpaired arm or by the therapist which gives ability for the subject to get involved in the decision of the grip aperture/closure. The values of $handopen\_button$ and $handclose\_button$ are binary values, which could be 1 when it is pressed and 0 when it is released. x_{start} is the starting position of the task. t and t_{hand} are the current time and the final time to complete hand opening, respectively. θ_{lower} and θ_{upper} represents the set of joint angles upper and lower limits, respectively and θ is set of the actual joint angles. τ_{robot} and $\tau_{rthreshold}$ are the torque applied to the motor of the robot and the torque threshold value, respectively. The torque applied to the motor of the hand assistive device and its threshold value is defined as τ_{hand} and $\tau_{hthreshold}$,

respectively. There is an emergency button on the controller of the robot and when it is pressed or released $emergency\_button$ value becomes 1 and 0, respectively. Additionally, a pause button is placed in front of the subject so that the subject can manually stop the task execution when he/she does not feel comfortable, or feel pain etc. When the pause button is pressed or released $pause\_button$ value becomes 1 and 0, respectively. In this application, $h_6(x)$, $h_7(x)$, $h_8(x)$ and $h_9(x)$ hypersurfaces are representing same region and all requires both arm assistive device and hand assistive device to be idle. Thus, we have 8 plant states which leads to 8 different regions and possible 256 (2^8) transitions. When a hypersurface is crossed than a plant event, which is simply an occurrence in the plant, is occurred. A plant event only causes a plant symbol to be generated. The following plant symbols are defined:

$\tilde{x}_1$: arm approaches to the bottle with the desired grip aperture,

$\tilde{x}_2$: arm reaches to the bottle,

$\tilde{x}_3$: hand reaches desirable grip closure to grasp the bottle,

$\tilde{x}_4$: arm leaves the bottle on the table,

$\tilde{x}_5$: hand reaches desirable grip aperture,

$\tilde{x}_6$: safety related issues happened such as robot joint angles are out of limits, or robot applied torque is above the threshold, or emergency button is pressed,

$\tilde{x}_7$: subject presses the pause button,

$\tilde{x}_8$: subject releases the pause button,

where $\tilde{X} = \{\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4, \tilde{x}_5, \tilde{x}_6, \tilde{x}_7, \tilde{x}_8\}$ is the set of plant symbols. The high-level controller needs to know which state was active before the pause button was pressed in order to provide the same task parameters to the subject when he/she resumes the task (when plant symbol $\tilde{x}_8$ is generated). Thus, a secondary plant event, which is called *SE1*, is defined in here to detect the previous state when the subject wants to continue with the task after he/she pauses. For example, when the subject presses pause button, a value is assigned to *SE1*. This value is retrieved when the subject resumes the task so that he or she can continue the therapy with the same task requirements. The secondary plant event, *SE1*, is defined as follows: if the subject presses pause button when he/she is approaching towards the bottle, then *SE1*=1 and a corresponding plant symbol $\tilde{x}_{81}$ is assigned. Similarly, if the subject presses pause button

when the subject is reaching to the bottle, or moving the bottle to mouth, or leaving the bottle on the table, or moving back to starting position (when only performing subtasks related to arm movement) then *SE1*=2, and the plant symbol $\tilde{x}_{82}$ is assigned. If the subject releases the pause button when he/she is closing his/her hand, then *SE1*=3 and a corresponding plant symbol $\tilde{x}_{83}$ is assigned. Finally, if the subject releases the pause button when he/she is opening his/her hand, then *SE1*=4 and the plant symbol $\tilde{x}_{84}$ is assigned.

In this application, the purpose of the DES controller (high-level controller) is to control the arm assistive device and the hand assistive device in a coordinated manner so that these devices are executed in the desired order without entering critical regions to complete an ADL task as required. The high-level controller is modeled as a discrete-event system (DES) deterministic finite automaton, which is specified by $D = (\tilde{S}, \tilde{X}, \tilde{R}, \delta, \phi)$. Here $\tilde{S}$ is the set of controller states. Each event is represented as a plant symbol, where $\tilde{X}$ is the set of such symbols, for all discrete states. The next discrete state is activated based on the current discrete state and the associated plant symbol using the following transition function: $\delta : \tilde{S} \times \tilde{X} \to \tilde{S}$. In order to notify the low-level assistive controllers the next course of action in the new discrete state, the controller generates a set of symbols, called control symbols, denoted by $\tilde{R}$, using an output function: $\phi : \tilde{S} \to \tilde{R}$. The action of the high-level controller is described in (1), where $\tilde{s}_i, \tilde{s}_j \in \tilde{S}$, $\tilde{x}_k \in \tilde{X}$ and $\tilde{r}_c \in \tilde{R}$. i and j represent the index of discrete states. k and c represent the index of plant symbols and control symbols, respectively. n is the time index that specifies the order of the symbols in the sequence.

$$\begin{aligned} \tilde{s}_j[n] &= \delta(\tilde{s}_i[n-1], \tilde{x}_k[n]) \\ \tilde{r}_c[n] &= \phi(\tilde{s}_j[n]) \end{aligned} \tag{1}$$

Each control state in the high-level controller captures one of the control actions which are the activation of only arm assistive device, or the activation of only hand assistive device, or the activation of both arm and hand assistive devices. Additionally, a memory state (such as a short-term memory [23] in our brain which stores information for a certain amount of time) could be used to detect which state was active before the pause button was pressed in order to provide the same task parameters when the subject resumes the task. When new actions are required for an ADL task, new states can easily be included in the set of the states, $\tilde{S}$. The high-level controller has six states $\tilde{S} = \{\tilde{s}_1, \tilde{s}_2, \tilde{s}_3, \tilde{s}_4, \tilde{s}_5, \tilde{s}_6\}$ which are:

$\tilde{s}_1$: hand device/arm device is active,

$\tilde{s}_2$: arm device is active,

$\tilde{s}_3$: hand device is active to close the hand,

$\tilde{s}_4$: hand device is active to open the hand,

$\tilde{s}_5$: arm/hand device is idle,

$\tilde{s}_6$: memory state.

The transition function $\delta : \tilde{P} \times \tilde{X} \to \tilde{P}$ uses the current state and the plant symbol to determine the next action that is required to update the rehabilitation task. For example, when the subject is reaching towards the target and a plant event is occurred that requires the arm device and the hand device to be stopped, then the transition function is used to transit from one active state, which activates the arm device, to another one, which stops both the arm and the hand assistive device, based on the plant event. The high-level controller generates a control symbol $\tilde{r}_c$, which is unique for each state $\tilde{s}_i$ using the output function $\phi(\tilde{s}_i) = \tilde{r}_c$ where $c \in \{1,2,3,4,5\}, i \in \{1,2,3,4,5,6\}$. The control symbols are defined as:

$\tilde{r}_1$: drive arm device to the bottle and drive hand device to open the hand,

$\tilde{r}_2$: drive arm device to the bottle, or move the bottle towards the mouth, or leave the bottle on the table, or go back to starting position,

$\tilde{r}_3$: drive hand device to close the hand to grasp the bottle,

$\tilde{r}_4$: drive hand device to open the hand to leave the bottle,

$\tilde{r}_5$: stop arm and hand device

where $\tilde{R} = \{\tilde{r}_1, \tilde{r}_2, \tilde{r}_3, \tilde{r}_4, \tilde{r}_5\}$ is the set of the control symbols. Inside the DES plant, the actuator converts the control symbols into continuous output and sends plant inputs to the low-level assistive controllers. The output of the control symbol used in here, which is called plant input, are given as: i) $if\ \tilde{r} = \tilde{r}_1\ \ then\ 1$, ii) $if\ \tilde{r} = \tilde{r}_2\ \ then\ 2$, iii) $if\ \tilde{r} = \tilde{r}_3\ \ then\ 3$, iv) $if\ \tilde{r} = \tilde{r}_4\ \ then\ 4$ and v) $if\ \tilde{r} = \tilde{r}_5\ \ then\ 0$. The control mechanism of the high-level controller is given in Fig. 3. The design of the elements of the DES plant and the DES controller is not unique and is dependent on the task, and the sensory information available from the rehabilitation robotic system.

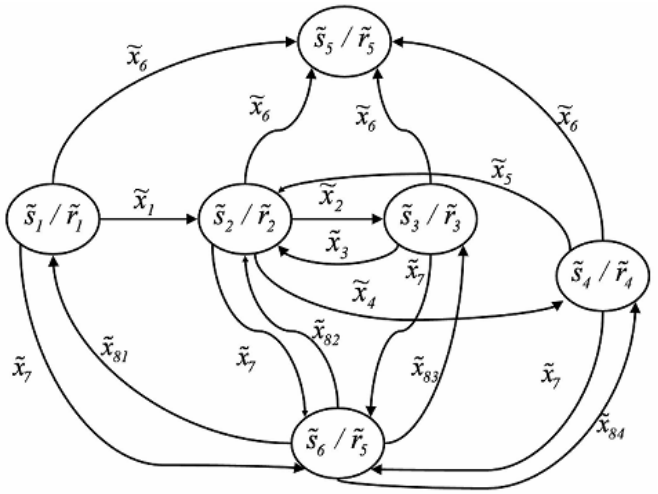


Figure 3 Control Mechanism for the High-Level Controller

6. Results

The subject is asked to wear the hand orthosis and then asked to place his forearm on the hand attachment device as shown in Fig. 2A. We have conducted experiments to demonstrate the efficacy of the control architecture when the task executed as planned and when an unplanned event happened during the execution of the ADL task. We had asked the subject to perform the DFB task which is described in details in Section V-A. Some of the subtasks of the DFB are shown in Fig. 4. The stateflow is used to implement the proposed high-level controller [24], where $\tilde{S} = \{\tilde{s}_1, \tilde{s}_2, \tilde{s}_3, \tilde{s}_4, \tilde{s}_5, \tilde{s}_6\}$ is described as *ArmHand*, *Arm*, *HandClose*, *HandOpen*, *Idle* and *Memory* respectively.

Figure 4 Subtasks of drinking from a bottle ADL task

In the first experiment the DFB task proceeded as planned. $ArmHand(\tilde{s}_1)$ became active and the control symbol $\tilde{r}_1$ was generated and the interface converted $\tilde{r}_1$ to $r = 1$ and both arm and hand assistive device were activated. The desired motion trajectories were given to the arm low-level assistive controller from *A* to *B* (Fig. 5-solid line). The discrete value 1 for

hand open (HO) (Fig. 6A-solid line) and 0 for hand close (HC) (Fig. 6B-solid line) was given to the hand low-level assistive controller from *A* to *B*. When $voltage\_ir$ became bigger than or equal to *1.85 Volts* that means subject reached two-thirds of the way of the reaching movement, and then $\tilde{x}_1$ was generated from the interface. $Arm(\tilde{s}_2)$ state became active, the control symbol $\tilde{r}_2$ was generated and the interface converted $\tilde{r}_2$ to $r = 2$ and only arm assistive device was activated (from *B* to *C* in Fig. 5-solid line and the discrete value 0 for both HO in Fig. 6A-solid line and HC in Fig. 6B-solid line). When the subject's position, x, is equal to the bottle position, x_{target}, then $\tilde{x}_2$ was generated, $HandClose(\tilde{s}_3)$ state became active, and the control symbol $\tilde{r}_3$ was generated and $r = 3$ was given to activate the hand assistive device for grasping the bottle (from *C* to *D* in Fig. 5-solid line and the discrete value 0 for HO in Fig. 6A-solid line and 1 for HC in Fig. 6B-solid line). The subject closed his fingers to grasp the bottle and when $voltage\_fsr\ (vsfr)$ dropped below the $voltage\_threshold$, which was selected as *4* Volts after few tests, $\tilde{x}_3$ was generated, then $Arm(\tilde{s}_2)$ state became active. When $Arm(\tilde{s}_2)$ state became active the control symbol $\tilde{r}_2$ was generated and then the interface converted $\tilde{r}_2$ to $r = 2$ which activated the arm assistive device again to assist the subject to move the bottle to his mouth, and drink water using a straw and then leave the bottle on the table (from *D* to *E* in Fig. 5-solid line and the discrete value 0 for both HO in Fig. 6A-solid line and HC in Fig. 6B-solid line). When the subject's position x is equal to x_{target}, which was selected as *0.29m* in this application, that means the subject left the bottle back on the table, $\tilde{x}_4$ was generated, $HandOpen(\tilde{s}_4)$ state became active and control symbol $\tilde{r}_4$ was given and the interface generated $r = 4$ to activate the hand assistive device to open his hand (from *E* to *F* in Fig. 5-solid line and the discrete value 1 for HO in Fig. 6A-solid line and 0 for HC in Fig. 6B-solid line). When the subject reached desirable grip aperture $(t == t_{hand})$, where t_{hand} was chosen as *5*s (this could be changed based on the therapist choice), then $\tilde{x}_5$ was generated and $Arm(\tilde{s}_2)$ state became active. When $Arm(\tilde{s}_2)$ state became active the control symbol $\tilde{r}_2$ was generated and the interface converted $\tilde{r}_2$ to $r = 2$ which activated the arm assistive device to go back to starting position (from *F* to *G* in Fig. 5-solid line and the discrete value 0 for both HO in Fig. 6A-solid line and HC in Fig. 6B-solid line).

In the second experiment, at some point of time (*t'*) during the task execution an unplanned event takes place. In this case, the subject wanted to pause for a while in the middle of the task and then reset the pause button (at *tt'*) when he was ready to complete the rest of the task. This scenario might represent when a stroke patient want to pause for a while due to some discomfort. The high-level controller dynamically and automatically generated the desired motion trajectories given in Fig. 5 and Fig. 6 as dashed lines by considering the subject's intention to pause the task. As it could be seen the subject's position at the time of the triggering (*tt'* in Fig. 5-dashed line) was automatically detected and taken as an initial position to continue the task where it was resumed with zero initial velocity. In such a case, if the high-level controller did not modify the desired trajectories to register the intention of the subject to pause the task, then the desired motion trajectories would start at point *tt'* with a different starting position and a non-zero velocity (Fig. 5-solid line), which could create unsafe operating conditions. Note that the subject was required to open his hand when the task restarted at *tt'*, however, the hand device would not have been activated at time *tt'* (as shown in Fig. 6A-solid line) if the high-level controller did not modify the desired trajectory. Additionally, it could also cause the subject to close his hand before reaching to target object at *ttt'* (from C to D in Fig. 6B-solid line). It is conceivable that one could pre-program all types of desired trajectories beforehand such that they could address all types of unplanned events, and retrieve them as needed. However, for non-trivial tasks, designing such a mechanism might be too difficult to manage and extend as needed. The presented high-level controller provides a systematic mechanism to tackle such issues. It could also be seen that the necessary desired trajectories and discrete values to activate/deactivate the arm and hand assistive devices could be created dynamically using the high-level controller. High-level controller monitored the progress of the task and made decision on the activation/deactivation of the assistive devices to complete the ADL task as required.

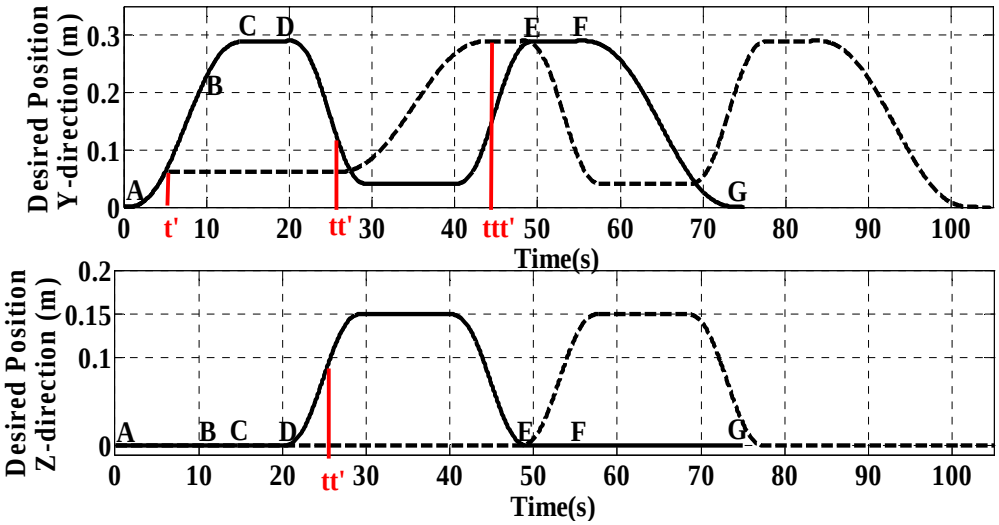


Figure 5 Desired Motion Trajectories Given to the Arm Low-Level Assistive Controller

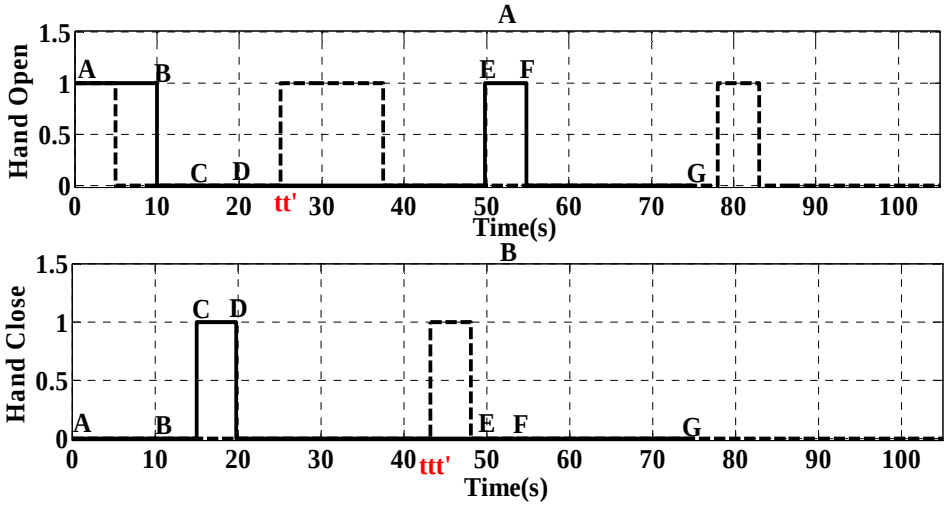


Figure 6 Desired Discrete Value (1/0) Given to the Hand Low-Level Assistive Controller

The desired trajectories given in Fig. 5 and Fig. 6 when the task executed as planned and when an unplanned event happened were generated and given to the arm low-level assistive controller and hand low-level assistive controller by the high-level controller so that these low-level assistive controllers provide assistance to the subject to complete the ADL task in the desired manner.

7. Discussion and Conclusion

In this paper we present a new control architecture that can coordinate both arm and hand assistive devices to help the subjects complete ADL tasks. The control architecture presented here is an example of a hybrid control system. There has been no work to our knowledge on designing similar type of control architecture for rehabilitation purposes. We briefly present the design of an arm low-level assistive controller and a hand low-level assistive controller which are used to provide assistance for the arm and the hand movement, respectively. Then

the high-level controller is presented which is used to coordinate these low-level assistive controllers in a safe manner to complete ADL tasks. We expect the proposed control architecture will give an opportunity for stroke patients to practice ADL tasks.

In this work the design of a high-level controller for the DFB task, which incorporates both the arm movement and the hand movement, is presented. Note that the proposed control mechanism described in Section 5.2 could be easily modified for the other ADL tasks such as eating using spoon, combing hair using a comb etc. Subjects are required to first reach the object, then grasp the object, move the object to his/her mouth or his/her hair, leave the object on the table and then move back to starting position. The task execution order for these ADL tasks is the same as drinking from a bottle ADL task. Thus, minor modifications such as i) the $voltage\_threshold$ value for the grasping spoon or comb could be changed to adjust the amount of grasping for various object sizes and properties, ii) t_{hand} value could be changed to vary the amount of time needed to release different objects. Additionally, task parameters could be adjusted not only for other ADL tasks but also for each patient for the same ADL task. For example, some patients may feel pain/uncomfortable when they grasp an object too much, thus $voltage\_threshold$ could be adjusted based on the patient's comfort and abilities. Moreover, the different movement abilities of stroke patients may require changes in the target object location, x_{target}. On the other hand some of the ADL tasks might not be in the exact order or as complicated as eating from the spoon or drinking from a bottle. New control mechanisms could be designed in the high-level controller using available actions (states), which were given in Section 5.2 for new ADL tasks. For example, a patient may be asked to perform to grasp and release an object for training of a hand movement which requires only opening hand to reach desirable grip aperture and closing hand to be able to grasp the object. Thus, only *HandClose* and *HandOpen* states are required to be coordinated in an order to execute the hand opening/closing activities as desired.

The hand device used in this paper does not allow the control of fingers independently in performing various hand rehabilitation tasks. However, this hand device is being used with C5 quadriplegic patients to perform their ADL tasks such as picking up bottle, holding things etc. The hand device allows opening/closing the hand with the adjustable power supply (fast and slow, low or stronger strength) and it is safe to be used if the right amount of pressure and speed is given to provide grasp function without damaging hand that lack sensation. We are also aware that a PUMA 560 robotic manipulator might not be ideal for rehabilitation applications. However the use of the hand attachment device, which has been described in

Section 3.1, provided a quick release mechanism to protect the subject's arm from injuries. Note that the presented control architecture is not specific to the presented arm and hand assistive devices but can be integrated with any other assistive devices.

Acknowledgment

We gratefully acknowledge the help of Dr. Thomas E. Groomes and therapist Sheila Davy of Vanderbilt University's Stallworth Rehabilitation Hospital for their feedback about the task design and Mark Felling for his feedback about the hand assistive device. The work was supported by Vanderbilt University Discovery grant.

References

[1] D. B. Matchar and P. W. Duncan, "Cost of Stroke," Stroke Clin. Updates, vol. 5, pp. 9-12, 1994.

[2] A. H. Association, "Heart and Stroke Statistical Update." vol. 2006.

[3] J. H. Cauraugh and J. J. Summers, "Neural plasticity and bilateral movements: A rehabilitation approach for chronic stroke," Prog Neurobiol., vol. 75, pp. 309-320, 2005.

[4] L. Ada, C. G. Canning, J. H. Carr, S. L. Kilbreath, and R. B. Shepherd, "Task-specific training of reaching and manipulation," in In Insights into Reach to Grasp movement. vol. 105, K. M. B. B. a. U. Castiello, Ed.: Elsevier-Biosoft, 1994, pp. 239-265.

[5] H. Ma and C. A. Trombly, "A synthesis of the effects of ocbottleational therapy for persons with stroke, Part II: remediation of impairments," Am J Ocbottle Ther, vol. 56, pp. 260-274, 2002.

[6] E. Taub, "Harnessing brain plasticity through behavioral techniques to produce new treatments in neurorehabilitation," American Psychologist, vol. 59, pp. 692-704, 2004.

[7] H. I. Krebs, M. Ferraro, S. P. Buerger, M. J. Newbery, A. Makiyama, M. Sandmann, D. Lynch, B. T. Volpe, and N. Hogan, "Rehabilitation robotics: pilot trial of a spatial extension for MIT-Manus," J Neuroengineering Rehabil, vol. 1, p. 5, 2004.

[8] P. S. Lum, C. G. Burgar, H. F. M. Van der Loos, P. C. Shor, M. Majmundar, and R. Yap, " MIME robotic device for upper-limb neurorehabilitation in subacute stroke subjects: A follow-up study," J. of Rehab. Research & Development, vol. 43, pp. 631-642, 2006.

[9] L. E. Kahn, M. L. Zygman, W. Z. Rymer, and D. J. Reinkensmeyer, "Robot-assisted reaching exercise promotes arm movement recovery in chronic hemiparetic stroke: a randomized controlled pilot study," J Neuroengineering Rehabil, vol. 3, pp. 1-13, 2006.

[10] R. Loureiro, F. Amirabdollahian, M. Topping, B. Driessen, and W. Harwin, "Upper limb mediated stroke therapy - GENTLE/s approach," Autonomous Robots, vol. 15, pp. 35-51, 2003.

[11] M. J. Johnson, K. J. Wisneski, J. Anderson, D. Nathan, and R. Smith, "Development of ADLER: The Activities of Daily Living Exercise Robot," IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, Pisa, Italy, 2006, pp. 254-259.

[12] D. Jack, R. Boian, A. S. Merians, M. Tremaine, G. C. Burdea, S. V. Adamovich, M. Recce, and H. Poizner, "Virtual reality-enhanced stroke rehabilitation," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 9, pp. 308–318, 2001.

[13] M. DiCicco, L. Lucas, and Y. Matsuoka, "Comparison of control strategies for an EMG controlled orthotic exoskeleton for the hand," in Proc. of IEEE International Conference on Robotics and Automation, 2004, pp. 1622 – 1627.

[14] Immersion Corporation, http://www.immersion.com/.

[15] T. Kline, D. Kamper, and B. Schmit, "Control system for pneumatically controlled glove to assist in grasp activities," in IEEE 9th International Conference on Rehabilitation Robotics, 2005, pp. 78 - 81.

[16] C. D. Takahashi, L. Der-Yeghiaian, V. H. Le, and S. C. Cramer, "A robotic device for hand motor therapy after stroke" in IEEE 9th International Conference on Rehabilitation Robotics, 2005, pp. 17 – 20.

[17] P. J. Antsaklis and X. D. Koutsoukos, "Hybrid Systems: Review and Recent Progress," in Software-Enabled Control: Information Technologies for Dynamical Systems, T. S. a. G. Balas, Ed.: IEEE Press, 2003, pp. 1-29.

[18] P. J. Antsaklis, "A Brief Introduction to the Theory and Applications of Hybrid Systems," Proceedings of the IEEE on Special Issue on Hybrid Systems: Theory and Applications, vol. 88, pp. 879-887, 2000.

[19] D. Erol and N. Sarkar, "Design and Implementation of an Assistive Controller for Rehabilitation Robotic Systems," Inter. J. of Adv. Rob. Sys, vol. 4, 2007.

[20] P. Culmer, A. Jackson, R. Richardson, B. Bhakta, M. Levesley, A. Cozens. "An admittance control scheme for a robotic upper-limb stroke rehabilitation system". In Proc. of International Conference on Engineering in Medicine and Biology society, pp. 5081 – 5084, 2005.

[21] M. A. Murphy, K. S. Sunnerhagen, B. Johnels, and C. Willen, "Three-dimensional kinematic motion analysis of a daily activity drinking from a glass: a pilot study." J Neuro. Rehabil. , vol. 16, pp. 3-18, 2006.

[22] M. Jeannerod, "Intersegmental coordination during reaching at natural visual objects," in Attention and performance IX, J. L. a. A. Baddeley, Ed. New Jersey, 1981, pp. 153-168.

[23] Short-term Memory: http://en.wikipedia.org/wiki/Short-term_memory.

[24] Stateflow, Mathworks Inc, http://www.mathworks.com/products/stateflow/?BB=1.

CHAPTER VI:

CONTRIBUTIONS AND FUTURE WORK

Contributions

The contributions of this dissertation are in the area of the intelligent controller design for robot-assisted rehabilitation systems. The main theoretical contributions of this dissertation are described as follows:

i) Design of an assistive controller that enhances smooth human-robot interaction between the robotic assistive device and the subject. The human arm parameters are estimated dynamically to provide robotic assistance in a subject-specific manner. The proposed assistive controller combines the benefit of system identification technique with the robustness of neural network-based methods

ii) Design of a new control approach to enable the coordination of both arm and hand assistive devices using a hybrid system modeling technique to help the subjects to complete their activities of daily living (ADL) tasks. Hybrid control framework has been effectively used in other fields, such as industrial robotics, medicine, and manufacturing. However, the presented control approach is the first of its kind that bring the benefit of hybrid system modeling in the field of the robot-assisted rehabilitation.

In order to validate the above controllers engineering contributions are made in the development of a robotic rehabilitation system. The main engineering contributions of this dissertation are given as follows:

i) Design of a rehabilitation task where the subjects not only make repetitive movement but also pay attention to the desired motion trajectory from visual feedback. This rehabilitation task was designed in such a manner that it required cognitive processing.

ii) Design of an experimental setup which includes a PUMA 560 robotic manipulator to provide assistance to the subject's arm movement, a hand attachment to attach the subject's hand to the robotic device and a hand assistive device to provide assistance to the subject's hand movement. This experimental setup gives ability for the subjects to perform the experiments in real-time by incorporating their feedback.

These contributions are expected to address some of the needs in the field of the rehabilitation robotics in terms of introducing advanced control methodologies.

Future Work

One of the future developments of this research is use of a functional magnetic resonance imaging (fMRI) procedure to investigate whether the presented task that included cognitive processing result in long-term brain reorganization. In Chapter IV, the artificial neural network (ANN) based proportional-integral (PI) gain scheduling controller has been evaluated and only the stiffness parameter has been used as the human arm parameter to train the ANN. It is also possible that mass and damping coefficients may be needed to better capture the arm characteristics of the stroke patients. In addition, the number and the nature of the features (e.g., here we used 3 features of the stiffness: the mean K_{μ}, the standard deviation K_{σ} and the maximum K_{max}) to train the ANN could be explored to investigate the improvement of the gain prediction. The presented results in Chapter IV were based on the training of the ANN with the data of 10 subjects. With the increase in training data set the prediction accuracy of the ANN will likely to be increased. As the robot interacts with more number of subjects, the training data sets will be increased. Another issue that may improve the performance of the ANN-based PI gain scheduling controller was to determine an optimal gain switching strategy. We used a fixed switching time for the control gains. However, it is possible to design a gain switching sequence that switches the predicted gains when a predefined threshold is exceeded. Additionally, it is possible to detect additional human state information to modify the task parameters using the control architecture described in Chapter III such as a voice recognition system to examine the patient's verbal commands or ECG signals to monitor the patients' heart rate to detect their exhaustion. This new state information could be used to change the difficulty level of the task and to make the necessary task modifications for safety of the patient. The presented control architecture in Chapter V is not specific to the presented the arm assistive device and the hand assistive device but can be integrated with any other assistive devices. The usability of the proposed control architecture can be tested with stroke patients.