SYNCHRONIZATION OF ARM AND HAND ASSISTIVE ROBOTIC DEVICES TO IMPART ACTIVITIES OF DAILY LIVING TASKS

Duygun Erol[1] and Nilanjan Sarkar[2]
[1]*Department of Electrical & Electronics Engineering, Yeditepe University, Kayisdagi, Istanbul, Turkey*
[2]*Department of Mechanical Engineering, Vanderbilt University, Nashville, TN, U.S.A.*

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

Abstract: Recent research in rehabilitation indicates that tasks that focus on activities of daily living (ADL) is likely to show significant increase in motor recovery after stroke. Most ADL tasks require patients to coordinate their arm and hand movements to complete ADL tasks. This paper presents a new control approach for robot assisted rehabilitation of stroke patients that enables them to perform ADL tasks by providing controlled and coordinated assistance to both arm and hand movements. The control architecture uses hybrid system modelling technique which consists of a high-level controller for decision-making and two low-level assistive controllers (arm and hand controllers) for arm and hand motion assistance. The presented controller is implemented on a test-bed and the results of this implementation are presented to demonstrate the feasibility of the proposed control architecture.

1 INTRODUCTION

Stroke is leading cause a disability that results in high costs to the individual and society (Matchar, 1994). Literature supports the idea of using intense and task oriented stroke rehabilitation (Cauraugh, 2005) and creating highly functional and task-oriented practice environments (Wood, 2003) that increase task engagement to promote motor learning, cerebral reorganization and recovery after stroke. The task-oriented approaches assume that control of movement is organized around goal-directed functional tasks and demonstrated promising results in producing a large transfer of increased limb use to the activities of daily living (ADL) (Ada, 1994). 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 movement. Consequently, robot-assisted rehabilitation that can quantitatively monitor and adapt to patient progress, and ensure consistency during rehabilitation has become an active research area to provide a solution to these problems. MIT-Manus (Krebs, 2004), MIME (Lum, 2006), ARM-Guide (Kahn, 2006) and the GENTLE/s (Loureiro, 2003) are the devices developed for arm rehabilitation, whereas Rutgers Master II-ND (Jack, 2001), the CyberGrasp (Immersion Corporation), a pneumatically controlled glove (Kline, 2005) and HWARD (Takahashi, 2005) are used for hand rehabilitation.

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 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. Robots that cannot simultaneously assist both arm and hand movements are of limited value in the ADL-focused task-oriented therapy approach. It is possible to integrate an arm assistive device and a hand assistive device to provide the necessary motion for ADL-focused task-oriented therapy. However, none of the existing controllers used for robot-assisted rehabilitation can be directly used for this purpose because they are not suited for controlling multiple systems in a coordinated manner. In this work, we address the controller design issue of a robot-assisted rehabilitation system that can simultaneously coordinate both arm and hand

Erol D. and Sarkar N. (2008).
SYNCHRONIZATION OF ARM AND HAND ASSISTIVE ROBOTIC DEVICES TO IMPART ACTIVITIES OF DAILY LIVING TASKS.
In *Proceedings of the Fifth International Conference on Informatics in Control, Automation and Robotics - ICSO*, pages 5-12
DOI: 10.5220/0001476700050012

motion to perform ADL tasks using an intelligent control architecture. The proposed control architecture uses hybrid system modeling technique that consists of a high-level controller and two low-level device controllers (e.g., arm and hand controllers). The versatility of the proposed control architecture is demonstrated on a test-bed consisting of an upper arm assistive device and a hand assistive device. Note that the presented control architecture is not specific to a given arm and hand assistive device but can be integrated with other previously proposed assistive systems.

In this paper, we first present the intelligent control architecture in Section 2, and then the rehabilitation robotic system and design details of the high-level controller are presented in Section 3. Later, results of the experiments to demonstrate the efficacy of the proposed controller architecture are presented in Section 4. Finally, the contributions of the work are presented in Section 5.

2 INTELLIGENT CONTROL ARCHITECTURE

Let us first present the proposed intelligent control architecture in the context of generic ADL tasks that require coordination of both arm and hand movement (e.g., eating, drinking, etc.). Stroke patients may not be able to complete the ADL tasks by themselves because of motor impairments. Thus, low-level arm assistive controller and low-level hand assistive controller may be used to provide assistance to the subject's arm and hand movement, respectively. 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 (e.g., completion of a subtask, safety related events etc.). A high-level controller (HLC) may be used to allocate task responsibility between the low-level assistive controllers (LLACs) based on the task requirements and specific events that may arise during the task performance. HLC 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 HLC and the LLACs may not communicate directly because each may operate in different domains. While the LLACs may operate in a continuous way, the HLC may need to make intermittent decisions in a discrete manner. Hybrid system theory provides mathematical tools that can accommodate both continuous and discrete systems in a unified manner. Thus, we take advantage of using a hybrid system model to design the proposed intelligent control architecture (Koutsoukos, 2000). In this architecture, the "Plant" represents both the assistive devices and their low-level assistive controllers and the Interface functions as analog-to-digital/digital-to-analog (AD/DA) adaptor.

The proposed control architecture for robot-assisted rehabilitation to be used to perform ADL tasks is presented in Fig. 1. In this architecture, the sensory information from the arm assistive device, the hand assistive device and the feedback from the human are monitored by the process-monitoring module through the interface. The sensory information (plant event) is converted to a plant symbol so that the HLC can recognize the event. Based on a plant symbol, the decision making module of the HLC sends its decision to the LLACs through the interface using the control symbols. Interface converts the control symbols to the plant inputs which are used to activate/deactivate the LLACs to complete the ADL task. The proposed control architecture is extendible in the sense that new events can be included by simply monitoring the new sensory information from the human and the assistive devices, and accommodated by introducing new decision rules.

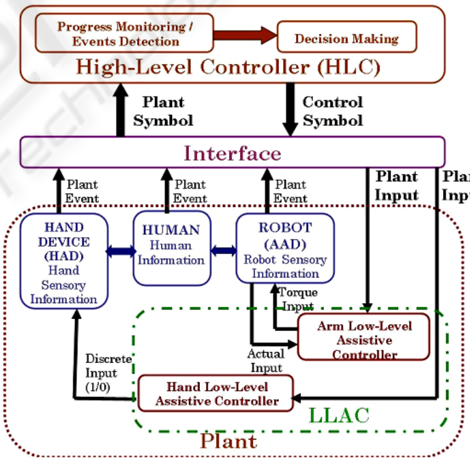

Figure 1: Control Architecture.

3 METHODOLOGY

The primary focus of this paper is to demonstrate how the assistive devices can be coordinated using the proposed intelligent control framework for a given ADL task. The intelligent control framework consists of a HLC and two low-level arm and hand assistive controllers. The focus of the paper is to design of HLC that can coordinate a number of given LLACs using the presented intelligent control framework. First we briefly present the rehabilitation system used as the test-bed to implement the intelligent control framework. We then present a detailed discussion on the design and implementation of the HLC and its workings within the presented control framework using an ADL task.

3.1 Rehabilitation Robotic System - a Test Bed

The rehabilitation robotic system used in this work consists of an arm assistive device, hand assistive device and two sensory systems (contact detection and proximity detection systems) (Fig. 2A). We have modified a Power Grip Assisted Grasp Wrist-Hand Orthosis (Broadened Horizons) as a hand assistive device (Fig. 2B). A computer control capability in a Matlab/Realtime Windows Target environment is added in order to integrate the hand device in the proposed control architecture. The subject is asked to follow the opening/closing speed of the hand device and if the subject cannot follow the hand device movement, then the hand device provides assistance to complement subject's effort to open/close his/her hand. The PUMA 560 robotic manipulator is augmented with a force-torque sensor and a hand attachment device (Fig.2A) to provide assistance to the upper arm movement (Erol, 2007). A proportional-integral-derivative (PID) position control is used as a low-level arm assistive controller for providing robotic assistance to a subject. The subject is asked to pay attention to tracking the desired position trajectory (visually monitoring his/her actual and desired position trajectories on a computer screen) as accurately as possible. If the subject deviates from the desired motion, then the low-level arm assistive controller provides robotic assistance to complement the subject's effort to complete the task as required. We have designed a contact detection system to provide sensory information about grasping activity that may be a part of an ADL task of interest. The force-sensitive resistors (FSR) (Interlink Electronics, Inc.) are placed on the fingertip to estimate the forces applied on the object during the grasping task (Fig. 2C). When the subject starts grasping an object, then the voltage across the FSR changes as a function of the applied force. Additionally, a proximity detection system (PDS) is designed in order to detect the closeness of the subject's hand relative to the object to be grasped. The PDS contains a phototransistor (sensitive to infrared light) and an infrared emitter, which are mounted onto two slender posts close to the object facing each other. When the subject approaches to the object by moving his/her hand between these posts, the continuity of the receiving signal (infrared beam) is broken and the corresponding voltage change is used to generate an event to inform the HLC that the subject is close to the object.

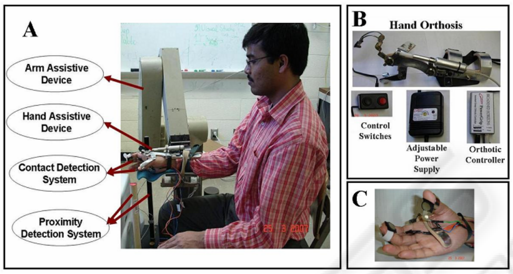

Figure 2A: Rehabilitation Robotic System, Figure 2B: Hand Assistive Device and Figure 2C: Force Sensor Resistors Placement on the Fingers.

3.2 Task Description

The main focus of this work is to present how a HLC is designed and how it functions within the proposed intelligent control architecture. The ADL tasks consist of several primitive movements such as reaching towards an object, grasping the object, lifting the object from the table, using the object for eating/drinking, and placing the object back on the table (Murphy, 2006) and they all require coordination between arm and hand movements. In here, we choose one of the ADL tasks, called drinking from a bottle (DFB) task to explain the HLC. We decompose the DFB task into the following primitive movements: 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, v) drink from a bottle using a straw, vi) place the bottle back on the table, vii) open the hand to leave the bottle back on the table and viii) go back to the starting position. Note that similar task decomposition could be defined for other ADL tasks.

3.3 Design Details of the High-Level Controller

3.3.1 Theory of Hybrid Control Systems

The hybrid control systems consist of a plant which is generally a continuous system to be controlled by a discrete event controller (DES) connected to the plant via an interface in a feedback configuration (Koutsoukos, 2000). If the plant is taken together with the interface, then it is called a DES plant model. The DES plant model is is a nondeterministic automaton which is represented 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 events; and $\tilde{R}$ is the set of control symbols. $\psi : \tilde{P} \times \tilde{R} \rightarrow 2^{\tilde{P}}$ is the state transition function. For a given DES plant state and a given control symbol, 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, $\lambda : \tilde{P} \times \tilde{P} \rightarrow 2^{\tilde{X}}$, maps the previous and current plant states to a set of plant symbols. The DES controller, which is called the HLC in this work, controls the DES plant. The HLC is responsible to coordinate the assistive devices based on both task and the safety requirements. The HLC 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 control states. Each event is converted to 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: $\delta : \tilde{S} \times \tilde{X} \rightarrow \tilde{S}$. In order to notify the LLACs the next course of action in the new discrete state, the HLC generates a set of symbols, called control symbols, denoted by $\tilde{R}$, using an output function: $\phi : \tilde{S} \rightarrow \tilde{R}$.

3.3.2 Modelling of an ADL Task using Hybrid Control System

Now we discuss how the above theory could be used to model and control an ADL task (e.g., the DFB task) for rehabilitation therapy. The first step is to design the DES plant and define the hypersurfaces that separates different discrete states. The hypersurfaces are used to detect the events and are decided considering the capabilities of the rehabilitation robotic systems and the requirements of the task. The following hypersurfaces are defined: $h_1 = vir > 0, h_2 = |x| \geq |x_t - \epsilon|$, $h_3 = (vfsr < vth) \wedge (hcb == 0)$, $h_4 = (|x| \leq |x_t - \epsilon|) \wedge (hob == 0)$, $h_5 = (t == t_{hand})$, $h_6 = \theta_l < \theta < \theta_u$, $h_7 = \tau_r \geq \tau_{rth}$, $h_8 = \tau_h \geq \tau_{hth}$, $h_9 = (eb == 1)$, $h_{10} = (pb == 1)$ $h_{11} = (pb == 0) \wedge (eb == 0)$, where vir is the voltage in the PDS system. x and x_t are the hand actual position and the object's position, respectively. ϵ is a value used to determine if the subject is close enough to the object's position. $vfsr$ and vth are the voltage across the FSRs and the threshold voltage, respectively. The values of hob and hcb are binary values, which could be *1* when it is pressed and *0* when it is released. t and t_{hand} are the current time and the final time to complete hand opening, respectively. θ_l and θ_u represent the set of lower and upper limits of the joint angles, respectively and θ is the set of the actual joint angles. τ_r and τ_{rth} are the torque applied to the motor of the arm assistive device 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_h and τ_{hth}, respectively. The values of eb and pb are binary values, which could be *1* when it is pressed and *0* when it is released. The above hypersurfaces can be classified into two groups: i) the hypersurfaces that are defined considering the requirements of the tasks (i.e., h_1-h_5), and ii) the hypersurfaces that are defined considering the capabilities of the rehabilitation robotic system (i.e., h_6-h_{11}). The hypersurfaces provide information to the HLC in order to make decisions for execution of the task in a safe manner. The set of DES plant states $\tilde{P}$ is based upon the set of hypersurfaces realized in the interface. Each region in the state space of the plant, bounded by the hypersurfaces, is associated with a state of the DES plant. During the execution of the task, the state evolves over time and the state trajectory enters a different region of the state space by crossing the hypersurfaces and a plant event, occurs when a hypersurface is crossed. A plant event generates a plant symbol to be used by the HLC. The plant symbols $\tilde{X}$ in the DES plant model $G = (\tilde{P}, \tilde{X}, \tilde{R}, \psi, \lambda)$ are defined as follows:

$$\tilde{x}[n] = \lambda(\tilde{p}[n-1], \tilde{p}[n]) \tag{1}$$

where $\tilde{x} \in \tilde{X}$, $\tilde{p} \in \tilde{P}$, λ is the output function and n is the time index that specifies the order of the symbols in the sequence. In (1) the plant symbol, $\tilde{x}$, is generated as an output function of the current and the previous plant state. We define the following plant symbols considering the hypersurfaces discussed before: i) $\tilde{x}_1$, the arm approaches to the bottle with the desired grip aperture, which is generated when h_1 is crossed, ii) $\tilde{x}_2$, the arm reaches to the bottle, which is generated when h_2 is crossed, iii) $\tilde{x}_3$, the hand reaches desired grip closure to grasp the bottle, which is generated when h_3 is crossed, iv) $\tilde{x}_4$, the arm leaves the bottle on the table, which is generated when h_4 is crossed, v) $\tilde{x}_5$, the hand reaches desired grip aperture, which is generated when h_5 is crossed, vi) $\tilde{x}_6$, safety related issues happened such as the robot joint angles are out of limits (when h_6 is crossed), or the robot applied torque is above its threshold (when h_7 is crossed), or hand device applied torque is above its threshold (when h_8 is crossed) or emergency button is pressed (when h_9 is crossed), vii) $\tilde{x}_7$, the subject presses the pause button, which is generated when h_{10} is crossed, and viii) $\tilde{x}_8$, the subject releases the pause button which is generated when h_{11} is crossed. $\tilde{X} = \{\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4, \tilde{x}_5, \tilde{x}_6, \tilde{x}_7, \tilde{x}_8\}$ is the set of plant symbols.

In this work, the purpose of the DES controller (HLC) is to activate/deactivate the assistive devices in a coordinated manner to complete the DFB task. In order to perform this coordination, the following control states are defined: $\tilde{s}_1$: both the hand device and arm device are active, ii) $\tilde{s}_2$: the arm device alone is active, iii) $\tilde{s}_3$: the hand device alone is active to close the hand, iv) $\tilde{s}_4$: the hand device alone is active to open the hand, v) $\tilde{s}_5$: both the arm and hand devices are idle. Additionally, a memory control state ($\tilde{s}_6$) is defined to detect the previous control actions when the subject wants to continue with the task after he/she presses pause button. $\tilde{S} = \{\tilde{s}_1, \tilde{s}_2, \tilde{s}_3, \tilde{s}_4, \tilde{s}_5, \tilde{s}_6\}$ is the set of control states in this application. When new control actions are required for an ADL task, new control states can be included in the set of the states, $\tilde{S}$. The transition function $\delta : \tilde{S} \times \tilde{X} \rightarrow \tilde{S}$ uses the current control state and the plant symbol to determine the next control action that is required to update the ADL task, where $\tilde{s} \in \tilde{S}$, $\tilde{x} \in \tilde{X}$, $\tilde{r} \in \tilde{R}$, and n is the time index that specifies the order of the symbols in the sequence.

$$\tilde{s}[n] = \delta(\tilde{s}[n-1], \tilde{x}[n]) \tag{2}$$

The HLC generates a control symbol $\tilde{r}$, which is unique for each state, $\tilde{s}$ as given in (3). In here, the following control symbols are defined: i) $\tilde{r}_1$: drive arm device towards the object while driving hand device to open the hand, ii) $\tilde{r}_2$: drive arm device to perform various primitive arm motion such as move the bottle towards the mouth etc., iii) $\tilde{r}_3$: drive hand device to close the hand to grasp the bottle, iv) $\tilde{r}_4$: drive hand device to open the hand to leave the bottle, and v) $\tilde{r}_5$: make arm and hand devices idle. The set of control symbols are defined as $\tilde{R} = \{\tilde{r}_1, \tilde{r}_2, \tilde{r}_3, \tilde{r}_4, \tilde{r}_5\}$.

$$\tilde{r}[n] = \phi(\tilde{s}[n]) \tag{3}$$

The LLACs cannot interpret the control symbols directly. Thus the interface converts the control symbols into continuous outputs, which are called plant inputs. The plant inputs are then sent to the LLACs to modify the ADL task. We define the following plant inputs: i) if $\tilde{r} = \tilde{r}_1$ then provide 1 to activate both arm and hand devices, ii) if $\tilde{r} = \tilde{r}_2$, then provide 2 to activate only the arm device, iii) if $\tilde{r} = \tilde{r}_3$, then provide 3 to activate only the hand device to close hand, iv) if $\tilde{r} = \tilde{r}_4$, then provide 4 to activate only the hand device to open hand, v) if $\tilde{r} = \tilde{r}_5$, then provide 0 to keep both arm and hand devices idle. Note that 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 robotic system.

4 RESULTS

In this section we present two experiments that were conducted to demonstrate the feasibility and usefulness of the proposed control architecture. Since we experiment with an unimpaired subject who could ideally do the DFB task by himself (unlike a real stroke patient) we instructed him to be passive so that we can demonstrate that the proposed control architecture was solely responsible for the coordinated arm and hand movements (which is the main objective of this work) as needed to complete the DFB task. Such an experimental condition is not only helpful to demonstrate the efficacy of the proposed control architecture but also could occur when a low functioning stroke survivor participates in a task-oriented therapy who will initially need continuous robotic assistance to perform an ADL.

The subject was asked to wear the hand device and then place his forearm on the hand attachment (Fig. 2A). In the first experiment (E1), we asked the subject to perform the DFB task, where the task proceeded as planned (i.e., there was no event occurred during the task that would require dynamic modification of the execution of the task; however, it still needed the necessary coordination between hand and arm motion). We designed a DFB task trajectory in consultation with a physical therapist as shown in Fig. 3. Fig. 3 shows how the DFB task was supposed to proceed: the subject was required to reach the bottle (A-B), grasp the bottle by applying a certain amount of force (B-C), bring the bottle to the mouth (C-D), drink water (D-E), bring back the bottle from where he picked it up (E-F), open hand to release the bottle (F-G), and then go back to the starting position (G-H). Furthermore, the desired trajectory from A-B into A-A' and A'-B trajectories have been decomposed because in naturalistic movement it has been shown that a subject reached his/her maximum aperture approximately two-third of the way through the duration of the reaching movement (Jeannerod, 1981).

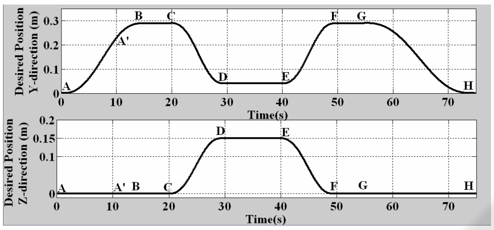

Figure 3: Desired Motion Trajectories for a DFB task.

The overall mechanism for the high-level control, which is used to activate/deactivate the LLACs, is shown in Fig. 4. When a device is active, we mean it tracks a non-zero trajectory and when the device is idle, we mean the device remains in its previous position set points.

Let us now explain how the HLC accomplished the DFB task using the control mechanism given in Fig. 4. When the DFB task started, $\tilde{s}_1$ became active where both arm and hand devices remained active till point A' to help the subject to open his hand while moving towards the bottle. Then at point A', the vir crosses a predefined threshold value, which confirmed that the subject reached close to the bottle ($\tilde{x}_1$ was generated) and $\tilde{s}_2$ state became active. The arm device remained active to help the subject to reach the bottle and the hand device was idle from A' to B. After that at point B, when the subject's position, x, was close to the bottle position, x_t, and then $\tilde{x}_2$ was generated and $\tilde{s}_3$ state became active. If $\tilde{s}_3$ was active, then the hand device remained active to assist the subject to grasp the bottle. Then, at point C, the $vfsr$ value was dropped below the threshold ($\tilde{x}_3$ was generated) and $\tilde{s}_2$ state became active again. The arm device remained active to assist the subject to move the bottle to his mouth, to drink water using a straw and at the end to leave the bottle on the table. When the subject brought the bottle back on the table at point F, $\tilde{x}_4$ was generated and $\tilde{s}_4$ state became active and the hand device remained active to help the subject to open his hand till G. Then the subject reached the desired grip aperture $\left(t == t_{hand}\right)$, $\tilde{x}_5$ was generated and $\tilde{s}_2$ state became active. The arm device remained active to help the subject to go back to the starting position. The actual trajectory of the subject was exactly same as the desired trajectory given in Fig. 3. The subject's hand configuration diagram was given in Fig. 5. It could be seen from the figures that the subject was able to track the desired trajectories while opening/closing his hand at desired times.

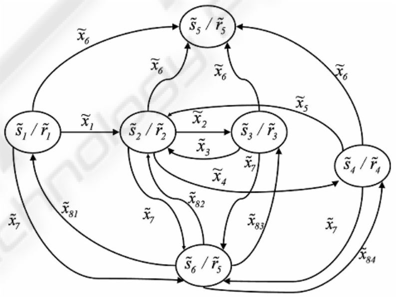

Figure 4: Control Mechanism for the HLC.

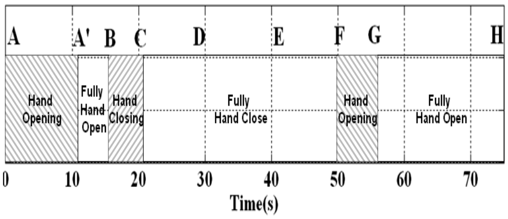

Figure 5: Hand Configuration Diagram for E1.

In the second experiment (E2), we demonstrated that if an event takes place at some point of time during the task execution that requires modification of the desired task trajectory such as a stroke patient wants to pause for a while due to some discomfort, then the HLC has the ability to dynamically modify the desired trajectory using the control mechanism given in Fig. 4. In this case, the subject started the execution of the task with the same desired trajectory as shown in Fig. 3 (which is the dotted trajectory in Fig. 6). During the execution of the task at time *t'*, the subject pressed the pause button at time *t'* (when $\tilde{s}_1$ was active), the plant symbol $\tilde{x}_7$ was generated and $\tilde{s}_6$ state became active and both the arm and hand devices became idle. $\tilde{s}_6$ state stored the previous active state. When the subject released the pause button at time *tt'* to continue the task execution, $\tilde{x}_{81}$ was generated and $\tilde{s}_1$ became active again to activate both the arm and hand devices to resume the task execution. The rest of the desired trajectory had been generated in the same way as it was described in E1 (Fig. 6-solid lines). It could be noticed from Fig. 6-solid lines that at time of *t'*, the assistive devices remained in their previous set points. Additionally, the subject's position at time of the *tt'* was automatically detected and taken as an initial position to continue the task where it was resumed with zero initial velocity (Fig. 6-solid lines). If the HLC did not modify the desired trajectories to register the intention of the subject to pause the task, then i) the desired motion trajectories would start at point *tt'* with a different starting position and a non-zero velocity (Fig. 6 -dotted lines), which could create unsafe operating conditions, and ii) the subject would close/open his hand at undesirable times. We had also noticed that the subject's actual trajectory was same as given in Fig. 6-solid line. Fig. 7 demonstrated the subject's hand configuration diagram for E2. It can be noticed that the subject was able to track the modified desired trajectory and he was able to coordinate his arm and hand motions in a safe and desired manner.

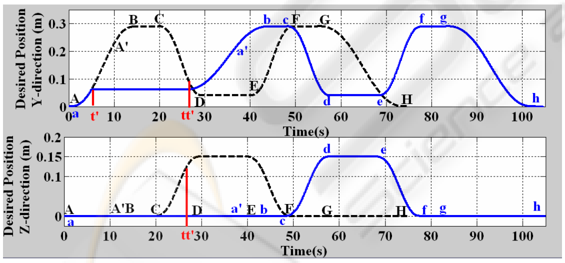

Figure 6: Desired Trajectory for the DFB task when an Unplanned Event Happened.

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.

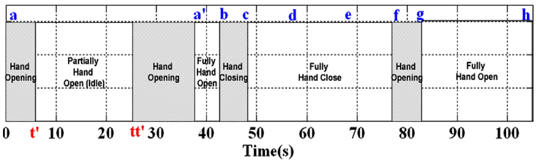

Figure 7: Hand Configuration Diagram for E2.

5 DISCUSSION/ CONCLUSIONS

The purpose of this work is to design a versatile control mechanism to enable robot-assisted rehabilitation in a task-oriented therapy that involves tasks requiring coordination of motion between arm and hand. In order to achieve this goal we design a new intelligent control architecture that is capable of coordinating both the arm and the hand assistive devices in a systematic manner. This control architecture exploits hybrid system modeling technique to provide the robotic assistance to enable a subject to perform ADL tasks that may be needed in a task-oriented therapy and has not been explored before for rehabilitation purpose. Hybrid system modeling technique offers systematic control design tools that provide design flexibility and extendibility of the controller, which gives ability to integrate multiple assistive devices and to add/modify various ADL task requirements in the intelligent control architecture. The control architecture combines a high-level controller and low-level assistive controllers (arm and hand). In here, the high-level controller is designed to coordinate with the low-level assistive controllers to improve the robotic assistance with the following objectives: 1) to supervise the assistive devices to produce necessary coordinated motion to complete a given ADL task, and 2) to monitor the progress and the safety of the ADL task such that necessary dynamic modifications of the task execution can be made to complete the given task in a safe manner.

Although the focus of the current work is to present a new high-level control methodology for rehabilitation that is independent of the low-level controllers, we want to mention the limitations of the hand and arm assistive devices used in the presented work. The hand device used in this paper does not allow independent control of fingers in performing various hand rehabilitation tasks. As discussed earlier, the focus of the paper is to present how arm and hand motion can be dynamically coordinated to accomplish ADL tasks. In that respect, the current hand device allowed us to perform an ADL task that showed the efficacy of the presented high-level controller. A more functional hand device would allow the patients to perform more complex ADL tasks. Note that this current hand device is being used with C5 quadriplegic patients to complete their ADL tasks such as picking up bottle etc. (Broadened Horizons). We are also aware that a PUMA 560 robotic manipulator might not be ideal for rehabilitation applications. However the use of safety mechanisms, both in hardware (e.g., emergency button, quick arm release mechanism etc.) and in software (e.g., within the design of the high-level controller) will minimize the scope of injuries. Note that the proposed control architecture is not specific to the presented assistive devices but can also be integrated with other assistive devices.

We believe that such a robot-assisted rehabilitation system with capabilities of coordination of both arm and hand movement is likely to combine the advantages of robot-assisted rehabilitation systems with the task-oriented therapy. In this paper, the efficacy of the proposed intelligent controller is demonstrated with healthy human subject. We are aware that a stroke patient with a spastic arm is much more different from a healthy subject following the robotic moves. In that respect, more functional assistive devices and their corresponding low-level controllers can be integrated inside the proposed intelligent controller to allow stroke patients to take part in task-oriented therapy. As a future work, it is possible to use intelligent robot-assisted rehabilitation systems in clinical trials to understand on how impairment changes carryover of gained functional abilities to real living environments and how robot-assisted environments influence these changes.

ACKNOWLEDGEMENTS

We gratefully acknowledge the help of Dr. Thomas E. Groomes and Sheila Davy of Vanderbilt University's Stallworth Rehabilitation Hospital for their feedback about task design and Mark Felling who is C5 quadriplegic patient for his feedback about the hand assistive device.

REFERENCES

Ada, L., Canning, C. G., Carr, J. H., Kilbreath, S. L. & Shepherd, R. B. (1994). Task-specific training of reaching and manipulation. *In Insights into Reach to Grasp movement*, *105*, 239-265.

Broadened Horizons, http://www.broadenedhorizons.com/

Cauraugh J. H., Summers, J.J. (2005). Neural plasticity and bilateral movements: A rehabilitation approach for chronic stroke. *Prog Neurobiol.*, *75*, 309-320.

Erol, D. & Sarkar, N. (2007), Design and Implementation of an Assistive Controller for Rehabilitation Robotic Systems. *Inter. J. of Adv. Rob. Sys*, *4*(*3*), 271-278.

Immersion Corporation, http://www.immersion.com/.

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

Jeannerod, M., (1981). '*Intersegmental coordination during reaching at natural visual objects'*. In Attention and performance IX, J. L. a. A. Baddeley, Ed. New Jersey, (pp. 153-168).

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. *J. Neuroengineering Rehabil*, *3*,1-13.

Kline, T., Kamper, D. & Schmit, B. (2005). Control system for pneumatically controlled glove to assist in grasp activities. *IEEE 9th International Conference on Rehabilitation Robotics*, pp. 78 - 81.

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. *J Neuroengineering Rehabil*, *1*, 1-15.

Koutsoukos, X. D., Antsaklis, P. J., Stiver, J. A. & Lemmon, M.D. (2000). Supervisory control of hybrid systems. *IEEE on Special Issue on Hybrid Systems: Theory and Applications*, *88*, 1026-1049.

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

Murphy, M. A., Sunnerhagen, K. S., Johnels, B. & Willen, C. (2006). Three-dimensional kinematic motion analysis of a daily activity drinking from a glass: a pilot study. *J. Neuroengineering Rehabil.*, *16*, 3-18.

Loureiro, R., Amirabdollahian, F., Topping, M., Driessen, B. & Harwin, W. (2003). 'Upper limb mediated stroke therapy - GENTLE/s approach. *Autonomous Robots*, *15*, 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. Res. & Dev.*, *43*, 631-642.

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

Wood, S. R., Murillo, N., Bach-y-Rita, P., Leder, R. S., Marks, J. T. & Page, S. J. (2003).Motivating, game-based stroke rehabilitation: a brief report. *Top Stroke Rehabil*, *10*, 134-140.