Task-specific Demonstration and Practiced Synergies for Writing with the ACT Hand

Eric Rombokas, Mark Malhotra, and Yoky Matsuoka

Abstract—Muscle synergies are hypothesized as a way to reduce the control space of redundant biological neuromuscular control. Robotics researchers are starting to use grasping synergies to simplify the task of coordinating many joints for complex manipulators. This is especially useful for hand movement control as there are many joints and muscles to control to achieve even simple tasks. This paper uses the Anatomically Correct Testbed (ACT) Robotic Hand to understand how task-specific synergies may be formed and used for complex robotic manipulation tasks such as writing. A comparison is made between synergies formed from (1) general hand movements, (2) task-specific demonstration of generic writing, and (3) practice of writing a specific letter. Results showed that performance using task specific demonstration synergies outperforms performance using general-purpose synergies in terms of completion time, energy expenditure, and trajectory error. Performance using practiced synergy outperforms the performance using demonstration synergies in trajectory error, even though there was no statistical difference in completion time and energy expenditure. These results indicate that task specific synergies from demonstration and practice allow a robotic hand to write better than using more generic synergies that may work for other tasks. How general/specific these synergies should be to optimize the performance of different complex tasks without learning too many specific synergies is an interesting topic for the future.

I. INTRODUCTION

Animals have available a great variety of effector activations capable of achieving a particular behavioral goal. Coordinating the activation of the musculature is not just a matter of knowing the effects of muscle activations on the output variable of interest, but also of combining individual activations into a single, coherent scheme [1]. We are interested in understanding how control redundancy is taken advantage of in biological systems to produce versatile movements while keeping the control strategy tractable. A greater understanding of biological control strategies may shed light on the importance of incorporating redundancy into artificial systems.

One of the goals of neuromuscular control research is to find motor primitives which can be combined to produce complex, task-specific movements. This idea has been widely expanded to include delays and other time dependance [3] [19], and a variety of methods for decomposition of observed behavior into synergy components. Related ideas are the "uncontrolled manifold" [4] and "minimum intervention" [6] hypotheses which argue that the nervous system uses the degrees of freedom which best suit the task constraints, and those solutions lie in the apparently constrained manifolds. There is also work demonstrating the use of unsupervised learning of system dynamics to build primitives based on sensorimotor experience in simulation [5].

In work by Santello et al [12] applying hand synergies to human grasping measurements, it was found that two principal components could account for >80% of the variance. These hand postural synergies greatly reduce the dimensionality from the original hand DoF space. Applying this to robotic grasping has been successful, as in the GraspIt! system [13], which simplifies the complex finger movement space into few control knobs. Recent exploration of synergies in grasping has shown their role in establishing force closure in grips [21] and has illustrated a procedure for translating from the kinematic space to the force domain. These results have focused on grasping, but we are interested in understanding synergies in the context of other dexterous manipulation tasks, such as writing.

The ultimate goals of the Anatomically Correct Testbed (ACT) hand robot are to understand how humans dexterously perform tasks and to identify simplified controls that act as a general purpose dexterous system. We want to use it as a tool to understand how complex manipulation tasks can be achieved with synergistic muscle control. When children learn how to write, it requires many years to learn to push into the paper at the right level of force and to produce smooth trajectories. During this process, teachers sometimes "hold" their hand and guide the desired movements. This is analogous to imitation-based learning that has been shown to improve robotic performance [7]. Other strategies in learning how to write letters include teachers providing a sheet with dotted letters that children trace over and over. This tracing may initially be done using pre-existing synergies, but this practice may allow formation of synergies more specific for...
the task.

In this paper, we took inspiration from these learning methods that children use to learn to write. To do so, we used the ACT hand to compare three different sets of synergies and performance in writing the letter ‘S.’ The three sets of synergies are: (1) general purpose synergies similar to Santello’s Eigen grasps (but corresponding to individual finger movements); (2) demonstration-based synergies generated when a teacher held the ACT hand and provided example motions suitable for writing in general; and (3) practice-based synergies generated from the ACT hand writing a specific letter ‘S’ using the general purpose synergies. The writing was performed using mostly finger movements while the wrist accommodated larger movements, and the arm was stationary. The control method is the same for each, so this experiment shows the differences in performance due to the method of synergy construction. The power of this approach is that the robot can encounter a completely novel situation, and construct appropriate synergies and fix controller parameters based only on experience of the task.

II. SYSTEM ARCHITECTURE

A. ACT Hand

The ACT Hand mimics the interactions among muscle excursions and joint movements produced by the bone geometries and tendon connectivity of the human hand. Each segment of the fingers is machined using human bone data, and are accurate in surface shape, mass, and center-of-gravity to the human equivalent. This mimicry results in a high-dimensional system with the redundancies and nonlinearities of the biological hand [7] [8]. The robot uses 24 motor-driven tendons to control the thumb, index, and middle fingers as well as the wrist. The motors are controlled through RTAI Linux at 200 Hz and have encoders with a resolution of 230 nm, allowing for precise control of each tendons length. The fingers and wrist have fewer degrees of freedom (2 wrist, 5 thumb, 4 index, 4 middle) than the number of motors, resulting in kinetic redundancy. The extensor mechanisms are complex webs of tendons on the dorsal side of the fingers, and are crucial for emulating dynamic human behavior [9]. As each tendon is pulled by a motor, it is routed through attachment points mimicking human tendon sheaths and following the contours of the bones. Since these bone shapes are complicated surfaces, the effective moment arm the tendon exerts on the joint varies with joint angle [10]. Some aspects of how the human nervous system exploits redundancy are understood [17] [18].

B. Tablet PC

A Toshiba Tecra M4 tablet PC measures stylus endtip position at an accuracy of .077 mm and transmits continuously over an ethernet TCP network at approximately 100 Hz to the ACT Hand control computer. This sensing approximates the human visual capability to see the ink markings resulting from pen manipulation.

![Mapping between actuation and task variables](image)

III. CONTROL METHODS

A motor controller acts as a means of mapping between the actuation variables we can control to the task variables which are manipulated to achieve the goal. In Section III-A we describe the variables used at each level of control, and in Section III-B we describe how the synergies and inverse models are constructed. In III-C we describe how control is achieved at each level. Section III-D describes how the trajectory is chosen.

A. Actuation and Task Variables

A hierarchical control paradigm was implemented using two linear functions, the synergy \( S(\delta l) \) and the inverse model \( F(\delta S) \). The tendon length changes \( \delta l \in \mathbb{R}^d \) are mapped to the intermediate representation \( \delta x \in \mathbb{R}^d \) by the synergy, where \( d \) is the number of synergy components being used, so the intermediate representation \( x \) is the synergy component response to the tendon lengths. The inverse model learns the synergy activations \( \delta x \in \mathbb{R}^d \) associated with pen tip motion \( \delta x, s \in \mathbb{R}^2 \).

Figure 2 illustrates the relationship between the representations. In the forward direction, changes in synergy response, \( \delta x \), are calculated from changes is tendon lengths, \( \delta l \).

\[
\delta x = S(\delta l)
\]

In the inverse direction, changes in synergy response are calculated from desired stylus displacements, \( \delta x \).

\[
\delta x = F(\delta s)
\]

B. Construction of Synergy and Inverse Model

While there are a variety of ideas about how to construct synergies, here we express the concept as a linear weighting of motor activations. Principal Component Analysis (PCA) reveals the dimensions which best account for the variability in \( l \). Choosing the components capturing at least 95% of the variance yields a lower-dimensional representation of the session. If the effective dimensionality of the task is low, then few components are necessary.

Non-linear mappings may be well approximated locally by linear ones. The benefit of a task-specific approach is that a small subspace of the reachable space is encountered, so the linear approximation is closer than it would be for a general-purpose model. For any stylus movement \( \delta s \) encountered in a session, there is a corresponding change in muscle lengths \( \delta l \) and the corresponding internal representation \( \delta x = S(\delta l) \). A linear fit to these observed stylus-posture pairs \( (\delta l, \delta x) \) is obtained using a least-squares optimization criterion and the result is \( F \) referred to above. \( F \) then, in the nomenclature of
neural control, is the inverse model mapping stylus motion (task variables) to synergy activation (controlled variables).

Consider a set of linear basis functions on stylus movements $g_i(\delta s): \mathbb{R}^2 \rightarrow \mathbb{R}^1$, where each basis function activates according to how much the desired stylus displacement moves in its preferred direction. Denote as $G(\delta s_k)$ the vector of all basis functions evaluated for example $k$. For $K$ total example pairs, we find the basis weightings $w$ which minimizes the squared error:

$$\arg\min_w \sum_{k=1}^K ||w^T G(\delta s_k) - \delta x_k||^2$$ (3)

$$F(\delta s_k) = w^T G(\delta s_k)$$ (4)

C. Controllers

A hierarchical controller architecture (Figure 3) was adopted to incorporate the constructed synergies and mappings into control of the robot. Each stage of the controller was tuned to account for its operating frequency and the unique features desired in its response.

At the low level, each of the 24 muscle lengths (cm) were tracked at 200 Hz with a PID controller on motor torque (mA) to provide fast response and steady-state error rejection. The integral term saturated at 75% of the max torque to prevent windup. These gains are conservative in order to guarantee stability even for large step inputs, thereby providing a layer of abstraction for the higher level controllers.

At the intermediate level, muscle lengths were mapped through the synergy $S(\delta l)$ to find the synergy component response, $\delta x$. A PI controller ran at 200 Hz to track component setpoints and its output was mapped into muscle length commands. Derivative control was not used because the input to the synergy controller comes through a ethernet network and its timing is not deterministic. Again, conservative gains ensure that the controller was stable even for large step inputs.

A high level tablet-space controller ran at a slower speed of 100 Hz, dictated by network speeds. Movements of the stylus endtip $\delta s$ were commanded based on the readout of current tablet position and the current desired position (See III-D).

The general-purpose, demonstration-based, and practice-based controllers were compared for gain settings which equalize the maximum speed of the stylus on the tablet. The power of an inverse model is defined as the sum over all possible directions of stylus movement.

$$P_T = \sum_{\delta s} F(\delta s)$$ (5)

This expresses the amount of change the inverse model calls for in the associated synergy, for all possible movement directions. Since the total power is proportional to the sum of each synergy’s power, a gain parameter is used to equalize the total actuation. Results for gains corresponding to three speed constants $c$ are reported here, denoted “fast”, “medium”, and “slow”.

$$K_{p,tablet} \cdot P_T = c$$ (6)

D. Trajectory Planner

The Synergy/Model architecture described in Figure 3 enables control of the stylus end tip to a desired point on the tablet. In order to follow a specified trajectory, a planner decides how quickly to proceed along the path and alters the setpoint for the tablet-space controller accordingly. The dynamics of the planner are determined by two parameters, the minimum success criterion, $s_{min}$, and the distance limit, $d_{max}$.

If the current stylus position is within a distance $s_{min}$ of a goal point on the trajectory, it is considered achieved and is removed from the trajectory. A goal point may also be achieved if the current position is closer to a later point in the trajectory. These rules encourage forward progress to improve operation speed at the cost of not guaranteeing completion of every goal point.

Rather than demanding instant completion of a goal, the planner institutes a maximum desired distance to travel, $d_{max}$. The tablet-space controller setpoint is never more than $d_{max}$ away from the current tablet position. Along with the tablet-space controller frequency and gain, this effectively caps the maximum command input to the synergy-space controller so that it is within its realm of stability. Assuming that this max command is near the limits of stability, setting $d_{max}$ to be
greater than $s_{\min}$ can ensure that the tablet-space controller is always running at its maximum capability.

IV. SYNERGY DECOMPOSITION

A. General-Purpose Controller

A general-purpose recording of muscle lengths in which each finger is moved through its range of motion provides the data used to build the baseline controller. Eight synergy components are required to account for 95% of the variability, consistent with abduction-adduction and flexion-extension for each finger and the wrist. Figure 4 shows the inverse models corresponding to each of the synergies. Direction $s_2$ moves the stylus tip toward and away from the hand, and $s_1$ moves left-right. The lobes indicate how much the synergy must activate to achieve motion in the stylus directions. For instance, the middle finger abduction synergy is activated primarily for motion in the $s_2$ dimension, positively for away-from-hand motion, and negatively for toward-hand motion.

B. Demonstration-Based Controller

A brief human-guided example of writing, in which the hand is passively moved through writing motions, provides the data used to build the demonstration-based synergies. These motions are not letters, but just "scribbling" in the task space. Only two synergy components are necessary to capture 95% of the variability encountered in the demonstration. The first roughly corresponds to movement in the $s_1$ tablet direction, and the second corresponds to movement in the $s_2$ tablet direction, as depicted in Figure 5.
C. Practice-Based Controller

The general-purpose controller writing the letter S five times is used to build practice-based synergies and inverse model. Three components are required to capture 95% of the variability. Note that the activations dictated by the general-purpose synergies (Figure 6) are more complex and multi-lobed, perhaps because they are encapsulating nonlinear effects.

V. RESULTS

The demonstration-based controller and practice-based controller outperform the general-purpose controller in time taken to complete (Figure 8), deviation from desired trajectory (y axis of Figure 7), and total energy expended (Figure 9), significant at a $>95\%$ confidence interval. This is interesting because these controllers use fewer synergy components than the general-purpose controller, and this result is consistent with improved human performance given task demonstrations and practice.

The practice synergies outperform demonstration synergies for trajectory error but not energy and time-to-complete. This improvement could be because the self-experience used to create this synergy captures control-relevant aspects of the task that the demonstration couldn’t provide. For instance, the tendons and joints have slightly different dynamic properties under active control than when the teacher is conducting the demonstration, or changes in pressure on the tablet surface could affect friction. It is likely that the practice synergy would exhibit less generalization to other letters than the demonstration synergy, since it only has experience of the letter ‘S’.

Figure 7 displays each trial as a point, and demonstrates clearly the benefit of the task-specific synergies. For a given writing speed, the practice-based synergy produces smaller average error, and both task-specific controllers improve the time taken to complete the letter.

VI. DISCUSSION

These results indicate that task specific synergies from demonstration and practice allow a robotic hand to write better than using more generic synergies that may work for other tasks. Building a library of such task-specific synergies could be a first step toward being able to accommodate a variety of dextrous manipulation tasks. This is consistent with neural control theories which suggest that multiple simultaneous internal models are used in the brain, rather than a single monolithic and complete model [23] [24].

An interesting topic for the future is how general/specific these synergies should be to optimize the performance of different tasks without learning too many specific synergies. It is also interesting to explore composition of multiple controllers; it is not clear that controllers and synergies built for different contexts can be combined linearly. However it is achieved, accounting for multiple tasks simultaneously would be a necessary requirement for a system interacting with the real world, and the more specific the synergies are, the more of them must be considered simultaneously. For instance, keeping the stylus pressing on the tablet surface with an appropriate amount of force is arguably a task in its own right, but is intimately tied to the stylus endtip position task. The results reported here were constrained in their maximum speed because the normal force was not being explicitly controlled so the stylus endtip could lift from the tablet under fast control, but normal force could be included as a task variable and considered as well.

Identifying task-relevant variables is the heart of understanding what constitutes a task, and affects the complexity of the synergies required. For instance, writing can be framed more temporally, with strokes being the fundamental space, instead of stylus position. In such a case, it might be appropriate to use time-varying synergies and models that take dynamics into account.
Fig. 9. Energy expended (Volts * Amps * Seconds). Each bar denotes the mean and Std. Dev. (N=5) using a particular controller/synergy, and speed constant. For a given speed constant (fast, medium, or slow), the demonstration-based and practice-based controllers use less energy.

REFERENCES