Studies on control methods for musculoskeletal robots using muscle synergies

Kin Chung Denny FU

September 2016

Studies on control methods for musculoskeletal robots using muscle synergies

A dissertation submitted to

The GRADUATE SCHOOL OF ENGINEERING SCIENCE

OSAKA UNIVERSITY

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY IN ENGINEERING

by

Kin Chung Denny FU

September 2016

Studies on control methods for musculoskeletal robots using muscle synergies

by

Kin Chung Denny FU

Submitted to the
The GRADUATE SCHOOL OF ENGINEERING SCIENCE
OSAKA UNIVERSITY
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY IN ENGINEERING

Abstract

Musculoskeletal robots have flexible and compliant structure inspired by biological creatures. They are capable of performing a variety of tasks, and can enhance dexterity and safety in various situations, such as replacing human jobs to perform dangerous and tedious tasks, and environments where robots work in close proximity with human. However, technical difficulties of controlling the complex structure that having many joints and muscles hinders development to practical applications. In biological studies, it has been suggested that the central nervous system of vertebrates simplifies control complexity by coordinating groups of muscle co-activations, namely muscle synergies, to produce movements, instead of controlling muscles independently. This research studies control methods using muscle synergies for musculoskeletal robots. First, in a case of controlling a musculoskeletal robot using an optimal control theory, analysis of several sets of muscle synergies arising from optimizing muscle activations according to different optimization objectives is carried out. Results show that muscle synergies can reduce control dimensionality while maintaining control performance. Moreover, the analysis demonstrates that the muscle synergies for performing a specific task can be extracted from muscle activations optimized according to an energy-related optimization objective function that does not include task-related variables. Second, the problem of how to extract muscle synergies given a data sample of parameterized muscle activations that are randomly initialized, without prior knowledge of robot dynamics is investigated; Most literature assumes that muscle synergies can be directly extracted from a given data sample of muscle activations that have inherent statistical regularities. A data preprocessing method is proposed to estimate a set of muscle activations that produces the same set of end-effector accelerations with minimum control efforts, from the randomly initialized parameterized muscle activations, based on system identification of the robot

dynamics using a kernel-based regression technique. A data-driven controller is also designed based on a sliding mode control technique to perform task space tracking control task. Results show that muscle synergies can be extracted from the estimated set of muscle activations, and can be utilized to control a musculoskeletal robot in a reduced control dimensionality. The proposed method contributes to enabling extraction of muscle synergies from data sample without statistical regularities. Third, the problem of enabling a musculoskeletal robot to obtain muscle synergies by itself is studied. Inspired by the motor skill learning in human infants, a data collection method is proposed based on a goal-directed exploration strategy. During exploration of designated targets spreading over an unknown task space, the robot is controlled in a reduced control dimensionality using muscle synergies and the data-driven task space tracking controller established from a local data sample. Results show that the proposed method can enable the robot to obtain muscle synergies and to establish a low-dimensional controller by itself, making a step forward to the development of autonomous musculoskeletal robots. Finally, this thesis concludes with several current limitations and future directions. The main contribution of this thesis is the investigation of the feasibility of control methods utilizing muscle synergies for a musculoskeletal robot. This research would be the first step to the realization of robots that can work in daily life.

Contents

1 Introduction **1**

1.1 Musculoskeletal robots 2

1.2 Muscle synergies 3

1.2.1 Interpretations of muscle synergies 3

1.2.2 Biological evidence 5

1.2.3 Relation to biological motor control 7

1.3 Related control methods in engineering 11

1.3.1 Optimal control theory 11

1.3.2 Task-space control 11

1.3.3 Learning-based control approach 12

1.4 Applications of muscle synergies in engineering 13

1.5 Research focuses 15

1.6 Thesis organization 17

2 Preliminaries **19**

2.1 Dynamics modeling of musculoskeletal robots 19

2.2 Extraction of muscle synergies 22

2.3 Control methods 23

2.3.1 Optimal control theory 24

2.3.2 Task-space tracking control 26

3 Analysis of muscle synergies and its utilization for generation of optimal movements **28**

3.1 Introduction . . . 29
3.2 Optimal control utilizing muscle synergies . . . 31
3.3 Muscle synergies with different properties . . . 33
3.3.1 Extracting muscle synergies . . . 33
3.3.2 Type 1: goal-related synergies . . . 34
3.3.3 Type 2: goal-unrelated synergies . . . 35
3.3.4 Type 3: Random synergies . . . 36
3.4 Performance Analysis . . . 36
3.4.1 Experiment 1: Feasibility of muscle synergies approach . . . 37
3.4.2 Experiment 2: Synergies with different properties . . . 41
3.5 Discussion . . . 48
3.5.1 The minimum number of synergies required . . . 48
3.5.2 Relationship of the cost-to-go values and the number of synergies used . . . 49
3.5.3 Determining the best number of synergies . . . 51
3.6 Summary . . . 53

4 Extracting muscle synergies from random movements for low- dimensional task-space control 55

4.1 Introduction . . . 56
4.2 Problem statement . . . 60
4.2.1 Collecting omnidirectional reaching-like movement data . . . 62
4.3 Estimation of nonlinear affine system . . . 64
4.4 Extraction of muscle synergies . . . 66
4.4.1 Definition . . . 67
4.4.2 Extraction of muscle synergies from estimated optimal control signals . . . 68
4.5 A sliding controller for overactuated system . . . 71
4.6 Inverse dynamics estimation . . . 75
4.6.1 Estimation of the inverse dynamics from optimal control signals 75

4.6.2 Regeneration of the same movements using lower-dimensional control signals . . . 75
4.6.3 Overall procedure . . . 77
4.7 Experiments . . . 79
4.7.1 Simulation setup . . . 79
4.7.2 Tracking a figure of 8 trajectory . . . 80
4.7.3 Dimension reduction on controlled movements and muscle synergies refinement . . . 85
4.8 Discussion . . . 88
4.8.1 The minimum number of synergies required . . . 88
4.8.2 Determining the best number of synergies . . . 89
4.9 Summary . . . 91

5 Obtaining muscle synergies in a goal-directed exploration scheme 92
5.1 Introduction . . . 93
5.2 The exploration scheme . . . 95
5.2.1 Notation definitions . . . 96
5.2.2 Overview . . . 98
5.2.3 The initialization process . . . 99
5.2.4 The try-to-reach process . . . 103
5.2.5 The stable point positioning process . . . 106
5.3 Experiments . . . 109
5.3.1 Exploration task . . . 110
5.3.2 Control performance . . . 113
5.4 Discussion . . . 116
5.4.1 On the choice of the number of synergies for exploration . . . 116
5.4.2 Time-invariant synergies vs time-varying synergies . . . 117
5.5 Summary . . . 118

6 Conclusion 119
6.1 Conclusion . . . 119

6.2 Limitations 121
6.3 Future work 122

A A human-like robotic arm simulation platform 125

A.1 The musculoskeletal structure 125
A.2 Kinematics of joints and end effector 128
A.3 Dynamics model 128

Chapter 1

Introduction

Musculoskeletal robots have flexible and compliant structure inspired by biological creatures. This structure imparts dexterity, flexibility and versatility to the robots. If the robots can be well controlled, these innate qualities can be brought out to contribute in various situations, especially in scenarios that require multitasking and safety. For instance, in tele-manipulation, a human-like musculoskeletal robots can be sent to dangerous or distant environments. The musculoskeletal structure provides possibilities of transferring operators' natural movements to effectively perform a variety of tasks [11]. The compliant musculoskeletal structure also can enhance safety when robots work with humans in close proximity, such as in after-stroke rehabilitation applications where an exoskeleton robot guides a patient's limb to follow some specific desired trajectories for therapy of movement recovery.

In considering control methods of musculoskeletal robots, learning from biological creature comes naturally in mind. This thesis concerns with a bio-inspired concept called muscle synergies; The central nervous system of vertebrates reduces control complexity by coordinating muscle synergies, instead of muscles independently. This research starts by analyzing performance of sets of muscle synergies with different inherent properties in a particular optimization problem, subsequently investigating how to obtain, and how to utilize muscle synergies in several engineering control problems, towards the ultimate goal of development of musculoskeletal robots that can work in daily life.

1.1 Musculoskeletal robots

Musculoskeletal robots have skeletal structure actuated by force-controllable actuators. The skeleton is constituted by connecting bones with artificial joints to provide a supportive structure for a robot, in contrast to vertebrates where bones are connected by ligaments [1]. The force-controllable actuators mimic the contraction mechanism of muscles in vertebrates, as opposed to conventional motors that provide rotary actuation at joints. Linear actuators such as pneumatic artificial muscles the McKibben muscles [2], electromagnetic linear actuators [3], or wire-driven type actuators in which wires attached to bones directly driven by motor [4] are examples of the force-controllable actuators.

The musculoskeletal structure is beneficial in various applications demanding for flexibility and safety. For instance, a human-like robotic arm can perform a task (e.g. holding an object) with different joint configurations. When there is sudden change (e.g. an obstacle in a place block some configuration), the robotic arm can still perform a task by changing to another admissible configuration. Within a close proximity or during interaction with human, the force-controllable actuators can be easily controlled or cut off to avoid undesired collision [5, 6] thus enhance safety. The musculoskeletal structure has biarticular actuation mechanism, where muscles actuate distal bones (muscles across two joints rather than one joint). The biarticular actuation mechanism has force output characteristics closer to human than conventional rotary actuation mechanism, enhances safety especially enhances in wearable robot applications [7]. A seven degree-of-freedom arm exoskeleton actuated by pneumatic artificial muscles for arm movement recovery training [8] is an example in rehabilitation application. Recently, efforts have been put in developing human-machine interface using electromyogram (EMG) signals to directly transfer operators' natural movements to remote robots in tele-operation applications [9, 10]. It is believed that musculoskeletal robots that mimic biological actuation mechanism are more suitable to transfer operators' dexterity [11].

However, the control of musculoskeletal robots is difficult. Musculoskeletal robots

usually have many actuators and many joints (degree-of-freedom). One difficulty is that computing control signals to achieve a desired task is generally an ill-posed problem, because the number of actuators is larger than the degree-of-freedom of the robots. It is also difficult to obtain accurate analytical models of the flexible and complicated musculoskeletal structure. This research particularly focuses on reducing the control dimensionality, taking inspiration of a biological motor control concept, namely muscle synergies.

1.2 Muscle synergies

How does the central nervous system (CNS) coordinate many muscles to produce movements and to perform various motor tasks? This is one of the fundamental questions in the study of biological motor control. Because of the redundancies of the joints and the muscles of musculoskeletal structure, there are many ways to accomplish a motor task. A motor task can be achieved by one of the many joint configurations, where each configuration can be attained by one of the many combinations of muscle activations. These redundancies cause a problem to the CNS because the task requirement provided is insufficient to select one of the infinite number of possible ways to accomplish the motor task [12]. This problem is known as the degree-of-freedom problem or the Bernstein's problem [13].

It has been suggested that the CNS simplifies control complexity by organizing control variables into modules [13, 14] such as spinal force field, kinematics strokes and muscle synergies [15]. By coordinating muscle synergies, the CNS produces a movement with fewer control variables; the CNS does not control each muscle independently [16,17]. This section gives a brief introduction about how muscle synergies can simply control complexity. Interpretations of muscle synergies are also mentioned.

1.2.1 Interpretations of muscle synergies

Muscle synergies are quantitatively studied by investigating statistical regularities in measurements of muscle activations. In biological studies, the measurements are

usually electromyogram (EMG) signals of motor tasks performed by a variety of species [18, 19]. Two components, muscle synergies and muscle synergy activations, are extracted from a given data sample of muscle activations. Common analyses assume that a given data sample can be approximated by linear combination of a set of muscle synergies. For the purpose of dimensionality reduction, it usually seeks for a set of synergies where the number of synergies is smaller than the number of muscles. The muscle activations are usually low-dimensional signals scaling the corresponding muscle synergies.

There are two main interpretations of muscle synergies, namely the time-invariant synergy and time-varying synergies. In the time-invariant synergies interpretation, each muscle synergy specifies a fixed pattern of muscle co-activations of a group of muscles. Time-invariant synergies are constant for all the time; they store spatial information of muscles and are task-independent [20]. Time-invariant synergies can be extracted by common linear matrix factorization tools. For instance, if principal component analysis (PCA) is used, the principal components are the time-invariant muscle synergies, and the corresponding scores are the synergy activations.

In the time-varying synergies interpretation, each muscle synergy specifies a sequence of muscle activations spanning for a particular duration of a group of muscles. Therefore, a synergy can be an input signal for actuation; it contains spatiotemporal information. The synergy activation defines the scale, time-lead/time-lag, and the duration of a synergy. The activations for a group of muscles are given by superposing time-varying synergies after modification by synergy activations. The activations can be either time-invariant or time-varying; This provides flexibility to adapt inherent regularities in a given data sample to provide better dimensionality reduction performance. The extraction of the time-varying synergies needs more complicated tools such as optimization process with specific constraints as demonstrated in [17].

Apart from the concept of muscles synergies, there is another interpretation of modular control mechanism in the CNS called the "uncontrolled manifold" [21], stating that the CNS coordinates elements (e.g. joint, muscles) that are task-related elements and leaves others elements uncontrolled. Across trails, a task-dependent uncontrolled

manifold can be observed from measuring the variance of all the elements. However, this concept requires a controller acting in high-dimensional space [22] (because all elements are controlled), which is different from the notion of control simplification in this study.

1.2.2 Biological evidence

Biological studies focus on validating the hypothesis of muscle synergies. One common approach is to obtain EMG signals from specific motor tasks of a certain species, followed by investigating the inherent statistical regularities; It is usually desired to identify muscle synergies and synergy activations that have lower dimensionality than the original number of muscles to approximate the acquired data sample.

To testify the muscle synergy hypothesis, various experiments have been carried out in a variety of species. In analyses of frog hindlimb movements such as reflexive motion [23], kicking [17], swimming, jumping, and walking [24], it has been reported that both the identified time-varying synergies [17] and time-invariant synergies [23] were directly related to the resulting kinematics characteristics. Further evidence was found in cat postural experiments [25, 26], where the time-invariant synergies obtained from the EMG signals from a set of natural postural configurations to maintain balance on a translating surface were consistent with that on a rotating surface, suggesting that the synergies captured specific biomechanical functionalities. In primates experiments, it was discovered that a small number of time-invariant synergies [27] extracted from a grasping task were able to reconstruct the EMG signals measured in other trials of the same task. A small number of time-varying synergies were also capable of accounting for a variety of grasping tasks, and adaptive to describe novel motor behaviors by tuning the scale and timing in the synergies [19].

The hypothesis of muscle synergies was also verified in human motor tasks. The EMG-signals of reaching tasks in different speeds and directions could be approximated by linear combinations of a set of extracted synergies; Similar synergies were found across subjects and with and without loading conditions [28, 29]. A similar finding was reported in [30], where a small number of time-invariant synergies could

explain the muscle activations in producing isometric forces by hand; The extracted synergies were correlated to a particular force direction that the synergy activations account for the amplitude of force. It has been also demonstrated that walking motions with different speeds and loading conditions could be explained by small number of time-varying synergies, which were found correlated to the kinematics of foot [31,32]

Not all experiments support the muscle synergy hypothesis. In an experiment of producing finger-tip force, it was found that the variance explained by each extracted synergies (by PCA) from the measured EMG signals has non-negligible fluctuation within trials, conflicting with the hypothesis that muscle activations are formed by small number of muscle synergies. [33] It has been also argued that the identified muscle synergies from EMG signals may be the consequence of task or biomechanical constraints, unrelated to the neural coupling of muscles in the CNS [34], although these results did not falsify the existence of a neural implementation of muscle synergies in the CNS.

More direct approach for testifying muscle synergy hypothesis has been conducted by trying to locate the neural implementation of muscle synergies in the CNS (e.g. motor cortex) when performing different motor tasks. Supportive evidence of muscles synergies has been found in cats that the sequential activations of specific groups of muscles were initialized and tuned by populations of neurons in the motor cortex [18]. Similar findings were reported in the study of the relationship between the neural activities in monkey's brain and muscle activations during pointing and reaching movements, where activations of groups of muscles that related to particular functionalities were correlated to the discharge of individual neurons in the primary motor cortex [35]. It was found that the time-invariant synergies extracted from the EMG signals resulting from micro-stimulations of particular regions in the motor cortex of two rhesus macaques were very similar to those identified from the reaching and grasping motions of the other rhesus macaques. In comparing time-invariant synergies extracted from the arm movements performed by healthy and that performed by brain-damaged patients, it was found that they are very similar, implying that the synergies were implemented in the unimpaired regions in the CNS [36]. In an

extension of the comparison to patients with more severe brain-damaged, the time-invariant synergies were found to be varied in the forms of preservation, merging and fractionation, indicating the CNS may response to the cortical damage [37]. A similar finding of preservation of synergy activations after stroke has also been reported in [38].

One of the limitations of analysis of measured EMG signals is that it is difficult to evaluate the feasibility of utilizing the extracted synergies to perform the observed motor tasks or generalized motor tasks. The validation of the modular control is usually carried out by reconstructing the observed data sample by a smaller number of muscle synergies as bases; However, the reconstructed muscle activities may not produce the same observed task [39]. Verifications using biologically plausible musculoskeletal model have been adopted to overcome this deficiency. A mathematical model of frog hindlimb was used in [40] in a synergies-based control scheme. It was shown that a low-dimensional dynamical model captures the natural dynamics of the frog hindlimb. Time-invariant synergies were obtained from data sample that was representative to account for both the low- and full-dimensional dynamics with minimum muscular effort. The synergies were found very similar to the synergies extracted from jumping and swimming motions of intact frogs. And the control performance of the low-dimensional control scheme using the proposed synergies was comparable with the full-dimensional controller that activated each muscle independently. An analysis was also conducted for human walking motion. It was reported that the time-invariant synergies extracted from the EMG signals of walking could be used as bases to reproduce walking kinematics and the ground reaction forces via a musculoskeletal model of human legs [41, 42], where the relative muscle activations and the synergy activations were optimized such that the difference between the experimental measurements and the forward simulation was minimized.

1.2.3 Relation to biological motor control

In addition the testification of the existence of muscle synergies, the relationship between muscle synergies and the act of control has been studied. In the presented

literature above, synergies were extracted from muscle activations of motor tasks, which are the consequence of the act of control by the CNS. This indicates there is a strong relationship between the existence of muscle synergies and the control strategies adopted in the CNS. Here two control strategies, task-oriented control strategy and optimal control strategy, that closely related to muscle synergies are introduced.

Internal models

How does the CNS produce muscle activations and movements? It has been suggested that the CNS uses internal models [43] to process sensorimotor information such as visual information, limb configurations, during motion planning, control, and learning [44–47]. Internal models that predict consequences of actions (motor commands) are known as forward models. For example, a forward model of arm dynamics can predict arm joint angles and velocities given current joint angles, velocities and motor commands [46]. Forward models have also been used to estimate unmeasurable quantities such as internal forces in ligaments for understanding energy utilization in walking simulation [48]. In contrast, internal inverse model acts as an controller, which transforms desired consequences to actions (i.e. motor commands that can achieve the desired consequences such as desired hand position trajectories) [49–51].

It has been suggested that muscle synergies simplify the representation and utilization of the internal models in the CNS by providing basis functions, thereby reducing the number of parameters to be processed in control [52]. For instance, for producing fast movements, an internal inverse model may be acquired to form an open-loop controller; Such controller can be synthesized by a small number of time-varying synergies [53]. Internal forward models provide estimations of the states and goal as feedback signals for error correction [54]. The error correction can also be achieved by coordination of muscle synergies [52, 55]. In a cat's postural task study [56], it was reported that the feasible force sets under the cat feet could be produced by a small number of time-invariant synergies using a simulated 3D static hindlimb model, suggesting that an internal model that produces postural force may coordinate time-invariant synergies.

Task-oriented control strategy

Task-oriented control strategy refers to the concept that the CNS focuses on achieving better control accuracy in terms of the task-space coordinates such as a position of a fingertip, rather than focuses on joint-space coordinates such as angles of shoulder, elbow and wrist [57]. The CNS represents limbs (joint space) and targets (task space) in different coordinates frames, and carries out transformation between the reference coordinates frame during execution of a movement [58]. A question about which coordinate frame (e.g. task-space coordinate frame which represents positions, a finger, or joint-space coordinate frame which represents angles of a shoulder, elbow and wrist of an arm) is used in the CNS during movement generation, has been mentioned in several literature [59, 60]. This concept has been investigated by experimental measurements of variance during movements (e.g. reaching movement). Because exerting control reduces error, the reference frame that revealed smaller variance would be more likely the central nervous system used in movement generation [61]. Several experimental studies also reported that variance in the task-space was smaller than the variance in joint space, either in both human [62] and animals [63]; This implies the CNS pays more attention to controlling the task space variables than the joint space variables.

Analyses have related muscle synergies with task-related variables to the performance of motor tasks. In [64], it was demonstrated that the EMG-signals of human reaching movements in different directions and speeds could be represented by a small number of time-varying synergies during the reaching movements and time-invariant synergies at the end of a reaching task (to maintain posture); The time-varying synergies were modulated in terms of the directions and speeds, implying that the task-relevant sensory information and the dynamics of the system could be incorporated into low-dimensional representation in the form of synergies to simply control. The functionality of muscle synergies in a human postural task was analyzed in [65]. In addition to the EMG signals of a person standing on a surface under perturbation, the task-related variable, which measured reaction forces to the

feet and the accelerations of the center of mass of the body, were included in the data sample for extraction of the so-called functional muscle synergies (time-invariant). It was found that the functional synergies extracted from one type of responses to the perturbations (non-stepping responses) were similar and could be used to reconstruct the EMG signals and the task-related variables of another type of responses (stepping responses), supporting the concept that muscle synergies are used to produce a predictable biomechanical function [25].

Optimal control strategy

It has been suggested that the CNS produces movements optimally – It selects an optimal control signal from the infinite number of solutions according to certain optimal principle in performing a motor task. In the field of computational study of motor control, an optimality principle called the minimum intervention principle [66,67], has been proposed to relate the act of the task-oriented control strategy and the resulting statistical regularities in control signals (e.g. EMG signals). It states that the optimization during generation of voluntary movements focuses on task-related control variables (e.g. specific groups of muscles that produce reaching motion in desired directions) and leaves task-unrelated control variables uncontrolled, which is related to the concept of uncontrolled manifold [14,21]. Using an optimal control theory to solve for solutions of a motor task [68–70], it has been demonstrated that a low-dimensional control space that reflects task-relevant dynamics of system is naturally identified [71]. Related results have been reported in [56,72]. Based on an anatomically-realistic musculoskeletal model of a cat, the muscle activations for keeping balance on a surface under translational perturbations were found by optimization constraining the forces and moments at the center of mass (task-related variables) while minimizing control efforts. It was found that the extracted time-invariant synergies could predict the EMG signals and the reaction forces on the surface observed experimentally, suggesting that muscle synergies mechanism can reduce control dimensionality and can achieve similar kinetics to the optimal solution.

1.3 Related control methods in engineering

This section briefly introduces several control methods in engineering for controlling musculoskeletal systems, especially those related to the computational aspects of control in vertebrates mentioned in section 1.2.3.

1.3.1 Optimal control theory

In the optimal control theory [73–75], the control problem is to find an optimal control law such that an objective function is optimized (minimized/maximized) while satisfying the robot dynamics itself, in contrast to common control problem that finding a closed-loop feedback control law such that the dynamics system is stable along with a given desired trajectory [76, 77]. Optimal control theory can solve for motion planning and control at the same time. For example, in realizing point-to-point arm reaching movement [78], there was no need to provide a pre-calculated desired trajectory. The objective function, or cost-to-go function [75], for a standard optimal control problem is composed of a cost at the final time (e.g. distance from a desired position at final time step) and accumulative cost over a finite/infinite time interval. The accumulative cost can be different from the definition of the cost at the final time, such as the accumulative amplitude of joint angular velocities and/or control efforts. In the biological studies, the optimal control theory is adopted to study different definition of the objective functions in producing movements, such as minimum jerk model [79] and minimum torque-change model [80].

1.3.2 Task-space control

In task-space control, or operational space control, the control task of a robot is to follow a give desired trajectory in task-space [81, 82] such as a desired robot end effector position trajectory. Since the dimensionality of the task space is usually lower than the joint space, there are infinitely many solutions (e.g. infinitely many combinations of joint torque) to achieve the same desired task space trajectory. A

task-space control law often consists of two components, a control term relating to the main task goal such as desired end effector accelerations, and a null space control term relating to a secondary goal such as joint stabilization [83] or posture control [84]. Essentially, task-space control laws can be implemented by learning-based approach. In [85], a task-space tracking control law for computing necessary torque of a simulated 7-DOF robotic arm to follow a figure of "8" trajectory was implemented, where the compute torque was a combination of locally estimated inverse dynamics models with weights computed by locally estimated forward dynamics models. In [86], a real 7-DOF robotic arm was controlled by online-updated inverse dynamics models estimated from local data in the vicinity of the current states of the robot.

1.3.3 Learning-based control approach

Techniques in machine learning allow implementation of controllers without knowing the robot structural parameters such as mass and link lengths, by utilizing forward models or inverse model estimated from experimental data. In the field of robotics, control algorithms are usually derived based on a dynamics model of a robot, which relates the control input (e.g. torque input at joints) and states (e.g. joint angles, velocities and accelerations) of the robot. The dynamic model can be analytically obtained by using standard methods such as the Euler-Lagrange equation of motion [87]. However, because of the structural variability and the uncertainty about physical parameters such as mass and link lengths, or because the states that fully define the dynamics may not be observable [73], it is usually difficult to obtain an exact dynamic model.

In model reference adaptive control [88, 89], a controller was updated based on the error between the desired states and the output of a forward dynamics model, which predicts the robot state in the next time step from an input action and a current robot state. Applications of forward models can also be found in solving optimal control problems such as model predictive control [90], in which an optimal action was computed by minimizing an objective function summing the prediction

error over a finite time step in the future; or in reinforcement learning [91], where the forward model gave probability of occurrence of next state given an input action and a current state. Inverse dynamics model, which gives actions (e.g. torque at joints) required to move the robot from current state to a desired state, can be found in various robotic application of such as computed torque control [76], where the inverse dynamics model gives the torque required for a robot to follow a desired trajectory (e.g. desired joint angles). If the inverse dynamics models exactly model the inverse dynamics of the robot, precise control performance can be achieved. More advanced techniques such as sliding mode control can offer tolerance for modeling inaccuracies and unmodeled dynamics [77].

1.4 Applications of muscle synergies in engineering

The modular control approach based on muscle synergies motivates robotic research to develop synergistic control strategies to reduce control complexity (in the sense of reducing the number of control variable) for high dimensional robotic systems. In contrast to biological studies that the objective is to justify (or falsify) the muscle synergies hypothesis, the objective in engineering is to develop controllers for accomplishment of a variety of tasks. This section highlights several works in robotic research that adopt the concept of muscle synergies.

One of the first synergies-based controllers was proposed in [92]. A control signal of the actuators was given by linear combination of time-varying synergies. Each synergy was defined by a single equilibrium point. This idea was inspired by a similar proposal in biological studies [93,94] that the CNS plans and executes limb movement as a temporal sequence of static attractor points for the limb. Various end-effector trajectories of a simulated planar kinematic chain could be produced by suitable choice of equilibrium points. Based on the same synthesis of synergies, a feedback controller that was able to drive a simulated 2D planar kinematic chain to synergy equilibrium position to follow the desired trajectory was proposed in [95]; The synergies were obtained from analytical solutions of an optimal control problem.

Obtaining muscle synergies from solutions of optimization problems can be found in [96, 97]. In [96], an analysis was carried out on a simulated planar robotic arm. Two sets of time-varying synergies extracted from optimal solutions of reaching tasks and via-point tasks solved by optimal control theory. Comparison of the two sets of synergies revealed that some synergies in the two sets were similar, suggesting that synergies arise regardless of the task context, and implying optimal motor behaviors can be efficiently generated by combinations of task-dependent and task-independent synergies. The existence of such compositional optimal control laws has been proven mathematically in [97]; For a certain class of stochastic optimal control problems that have a particular form of the optimization objective function called the cost-to-go function in defining a task, an optimal control law that is a linear combination of some functions. These functions are the solutions of other optimal control problems and can be represented as time-varying synergies (or primitives), although the acquisition of such time-varying synergies is not provided.

The acquisition of time-varying synergies without given an system dynamics model has been demonstrated in [98]. In the proposed hierarchical control scheme, muscle synergies translate high-level control signals encoded in low-dimensionality to actual muscle activations, via some internal variables that receive sensory signals; There exists inverse model that maps the sensory signals to the muscle synergies. The inverse model, as well as the time-varying synergies, can be learned from observed data sample, and form a low-dimensional controller. However, whether the controller can perform generalized tasks have not yet been testified. Reinforcement learning can solve optimal control problem adaptively without given system dynamics [91, 99]. Under the reinforcement learning framework, a composite control law is defined as a linear combination of time-varying synergies; Each synergy is a parameterized control policy. A given task is achieved by solving an associated Markov decision process to determine optimal parameters in the composite control law that maximizes the expected reward. It was shown that the introduction of time-varying synergies facilitate learning novel control policies, in a scenario that required a simulated muscle-actuated planar robot to complete reaching tasks in the present of obstacles.

One advantage of the time-invariant synergies is that they are simpler. Comparing with the time-varying synergies, they enable easier implementation of existing feedback control techniques, since the time-invariant synergies encode fixed muscle co-activations (spatial information) that a low-dimensional controller generates synergy activations. Although encoding temporal information in the time-varying synergies provides good dimensionality reduction performance, it is also more difficult to implement feedback controller [20].

Taking the advantage of simplicity, feedback controllers based on time-invariant synergies have been implemented in several works in robotic research. In the development of the tendon-driven robotic ACT hand [100], time-invariant synergies were adopted in a PID feedback controller that controled a finger-tip to follow a circular trajectory on a virtual plane accurately. In addition to the use of muscle synergies to reduce control dimensionality, the sensory signals were adopted to reduce the observation space, leading to a low-dimensional dynamical system where the feedback controller was derived. Without the knowledge of robot dynamics, a learning-based control scheme has been proposed in [101] to obtain muscle synergies using unsupervised Hebbian-like algorithm and to learn a low-dimensional feedforward controller based on supervised learning techniques; In the experiment of a single-joint robot driven by four tendons connecting to independent motors, the time-invariant muscle synergies were obtained from a data sample of robot responses of spontaneous single muscle twitches with fixed amplitude and duration. The low-dimensional controller was learned to minimize task error. In contrast to most literature where synergies have been obtained from a data sample of movements with specific task goals, this work demonstrates that time-invariant synergies can also be obtained from a data sample that is not generated with specific task goals.

1.5 Research focuses

This thesis puts focuses on the extraction and utilization of time-invariant synergies. The objectives of research in robotics should focus on developing control

methods that allow robotic systems to achieve a variety of tasks. Adopting time-invariant synergies is a straightforward approach, also because they are simpler to extract, and allow implementation of existing feedback control methods. Although time-varying synergies are more flexible that may enhance dimensionality reduction performance for specific data regularities, they require more complicated methods for extraction and control. This thesis consists of three technical parts:

1. A feasibility study comparing muscle synergies arisen from different optimization objective criteria in producing voluntary movements.

2. A method for extracting muscles synergies from movements actuated by randomly parameterized control signals, and derivation of a task-oriented controller utilizing the extracted muscle synergies, without the need of known analytical model of a robot.

3. A data collection method that a robot can gather appropriate data sample by itself for extraction of muscle synergies.

The first part investigates which optimization objective criteria are suitable definitions for extracting muscle synergies. Muscle synergies have to be extracted from data source before utilization for controlling musculoskeletal robots. In most literature, the data source is given and it is assumed that muscle synergies can be extracted. As mentioned in section 1.2.3, correlation (statistical regularities) between muscle activation, and thus the existence of muscle synergies, is closely related to the optimization process by the central nervous system. A suitable definition of the objective function in the optimization of voluntary movement is important to the development of extraction method.

Followed by the feasibility analysis, the second part focuses on developing a method to extract muscle synergies without given analytical dynamics models of the robot. In particular, extraction method from a data sample random movements is of special interest, since it is usually easier to generate random movements. The extraction method is developed based on the results obtained in the first part. A

learning-based approach is adopted, where an estimation technique was formulation in order to estimate the dynamics models of the robot. The estimated forward dynamics model contributes to the extraction of the muscles synergies, and an estimated inverse dynamics model is used for control. A task-oriented control technique is also derived for a musculoskeletal robot to track a desired trajectory of the end effector position in task space.

Finally, the third part describes a data collection method which adopts goal-directed exploration strategy. Towards the development of autonomous musculoskeletal robots, it is important to equip robot with self-learning ability. Inspired by the efficient motor skill learning strategy by means of goal-directed exploration, a method is proposed such that a musculoskeletal robot can collect data sample by trying to reach pre-defined targets spreading over the task space successively. Using the collected data sample, muscle synergies, the forward and inverse dynamics models of the robot are obtained such that the robot is controlled in reduced dimensionality during exploration. The proposed method enables a robot to gather data sample for extraction of muscle synergies by itself, paving a way for the development of autonomous musculoskeletal robots that can work in daily life.

1.6 Thesis organization

Chapter 2 gives preliminaries. A musculoskeletal model used in this thesis is described. Technical details of control methods, an estimation techniques and pattern recognition tools for extracting synergies are provided.

Chapter 3 analyzes muscle synergies arisen from optimization according to several objective criteria in generating voluntary movements. The optimization processes, together with a definition of optimization objective criteria for generating data source for extracting muscle synergies, and control method for producing the voluntary movement are described in detail.

Chapter 4 proposes the method for extracting muscle synergies without the need of given data source and dynamics models of the robot. The procedure for extracting

muscle synergies is elaborated. Estimation formulation and a task-space tracking controller for a class of musculoskeletal systems are derived.

Chapter 5 presents the data collection method for extraction of muscle synergies where the robot can gather data sample by itself, by adopting goal-directed exploration strategy. The method is developed based on the method presented in Chapter 4. Detailed implementation is provided.

Chapter 6 concludes this thesis and provides the future plan.

Chapter 2

Preliminaries

2.1 Dynamics modeling of musculoskeletal robots

A musculoskeletal robot is actuated by contraction forces provided by actuators (muscles) attached to the skeleton. Generally, the dynamics of a musculoskeletal robot in joint-space can be described by conventional rigid-body equation of motion:

$$M\Big(\boldsymbol{x}(t)\Big)\ddot{\boldsymbol{x}}(t) + C\Big(\boldsymbol{x}(t), \dot{\boldsymbol{x}}(t)\Big)\dot{\boldsymbol{x}}(t) + G\Big(\boldsymbol{x}(t)\Big) = \boldsymbol{\tau}\Big(\boldsymbol{x}(t), \boldsymbol{u}(t), t\Big) \tag{2.1}$$

where $\boldsymbol{x}(t) \in \Re^{n^x}$, $\dot{\boldsymbol{x}}(t) \in \Re^{n^x}$ and $\ddot{\boldsymbol{x}}(t) \in \Re^{n^x}$ are the joint angles, joint velocities and joint accelerations, respectively. t is the time. This time argument will be dropped in the later context for compact notation unless necessary. $M\Big(\boldsymbol{x}\Big)$ is the inertia matrix. Multiplying the matrix $C\Big(\boldsymbol{x}, \dot{\boldsymbol{x}}\Big) \in \Re^{n^x \times n^x}$ by $\dot{\boldsymbol{x}}$ yields the centrifugal and Coriolis forces. $G\Big(\boldsymbol{x}\Big) \in \Re^{n^x}$ is the gravity term. The control input to the actuators are constrained to be nonnegative and upper bounded $\mathbf{0} \leq \boldsymbol{u}(t) \leq \boldsymbol{u}^{ub}$.

Musculoskeletal robots are usually overactuated systems where the number of actuators is larger than that of the joints $n^u \geq n^x$. The actuators provide $\bar{j}$ contraction forces $F\Big(\boldsymbol{u}, t\Big) = \Big[\vec{f}_1, ..., \vec{f}_{\bar{j}}\Big] \in \Re^{n^y \times \bar{j}}$ when applying control input $\boldsymbol{u}$ at time t, where each column of $F\Big(\boldsymbol{u}, t\Big)$ is a force vector in the fixed global coordinate frame $\sum_{global}$. $n^y = 2$ and $n^y = 3$ if the robot works in a two and three dimensional task space, respectively. $\boldsymbol{\tau}\Big(\boldsymbol{x}, \boldsymbol{u}, t\Big) \in \Re^{n^x}$ is the resulting torque at the joints when applying

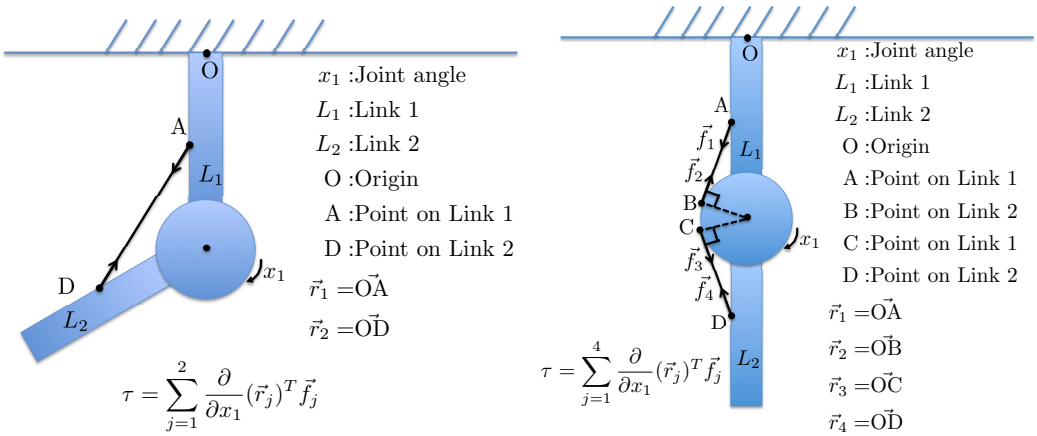


Figure 2-1: A 2-link robot actuated by one muscle. The two links are represented by the two rods and the revolution joint is depicted as the circle. (a) When there is no muscle wrapping, joint torque is produced by 2 forces exerting at the two muscle attachment points. (b) When there is muscle wrapping, joint torque is produced by 4 forces exerting at the two muscle attachment points (A and D) and the two tangent points at the joint.

control input $\boldsymbol{u} \in \Re^{n^u}$ at joint angles $\boldsymbol{x}$ at time t. The torque $\boldsymbol{\tau}$ is linearly related to the contraction forces F:

$$\boldsymbol{\tau}\left(\boldsymbol{x}, \boldsymbol{u}, t\right) = \sum_{j=1}^{\bar{j}} \Xi_j\left(\boldsymbol{x}\right) \vec{f}_j \tag{2.2}$$

where $\Xi_j\left(\boldsymbol{x}\right) \in \Re^{n^x \times n^y}$ is a matrix depends on the positions where the forces $\vec{f}_j$ exert to the skeleton. Let $\vec{r}_j$ be a position vector where $\vec{f}_j$ exerts at with respect to the global coordinate frame $\sum_{global}$, the rows of Ξ_j are the partial derivatives of with respect to the joints $\boldsymbol{x}$:

$$\Xi_j\left(\boldsymbol{x}\right) = \left[\frac{\partial}{\partial x_1}\vec{r}_j, ..., \frac{\partial}{\partial x_{n^x}}\vec{r}_j\right]^T \tag{2.3}$$

where T denotes the transpose operation. The number of forces $\bar{j}$ depends on the configuration $\boldsymbol{x}$ of the robot. More forces are exerted to the skeletal when muscle wrapping at the joints occurs. Fig. 2-1 depicts an example of a 2-links robot actuated by a muscle for the cases without and with muscle wrapping at the joint.

The overall dynamics depends on the characteristics of the actuators that how the contractile force $\vec{f}_j$ relates to the control input u_j. Throughout this thesis, actuators

with contraction forces linearly related to the control input without time delay are considered:

$$\|\vec{f_j}\| = c_j u_j \tag{2.4}$$

where c_j is a nonnegative scalar specifying the maximum amplitude of the force that the actuator j can produce. Equation (2.4) can model a simple muscle that has inextensible tendon such as the rigid-tendon models in [102]. Because this research focuses on investigating how muscle synergies can reduce control complexity by dimensionality reduction, the above simple linear muscle model (2.4) is adopted. Common muscle models having dynamics of contraction [103, 104] that introduce unobservable states into the robot dynamics are out of the scope of the present work.

The dynamics of a musculoskeletal robot with actuators having linear input-output relationship (2.4) can be described by the following nonlinear equations where the equation of motion is affine in control $\boldsymbol{u}$:

$$\begin{aligned} \ddot{\boldsymbol{x}}(t) &= f\Big(\tilde{\boldsymbol{x}}\Big) + g\Big(\tilde{\boldsymbol{x}}\Big)\boldsymbol{u} \\ \boldsymbol{y} &= h\Big(\boldsymbol{x}\Big) \\ &\vdots \\ \ddot{\boldsymbol{y}}(t) &= \alpha\Big(\tilde{\boldsymbol{x}}\Big) + \beta\Big(\tilde{\boldsymbol{x}}\Big)\boldsymbol{u} \end{aligned} \tag{2.5}$$

where $\tilde{\boldsymbol{x}} = [\boldsymbol{x}^T, \dot{\boldsymbol{x}}^T] \in \Re^{2n^x}$, $\boldsymbol{y} \in \Re^{n^y}$ is the position of the end effector with respect to the fixed global frame $\sum_{global}$. The muscle activation, i.e. the control input $\boldsymbol{u}$, are nonnegative and bounded such that $\boldsymbol{0} \leq \boldsymbol{u}(t) \leq \boldsymbol{u}^{ub}$. $f\Big(\tilde{\boldsymbol{x}}\Big) \in \Re^{n^x}$, $g\Big(\tilde{\boldsymbol{x}}\Big) \in \Re^{n^x \times n^u}$ and $h\Big(\tilde{\boldsymbol{x}}\Big) \in \Re^{n^y}$ are nonlinear functions obtained by substituting (2.4) and (2.2) followed by rearranging (2.1). The last equation in (2.5), which is the dynamics in task space, is obtained by differentiating the second equation twice with respect to time t followed by eliminating the term $\ddot{\boldsymbol{x}}$ using the first equation. $\alpha\Big(\tilde{\boldsymbol{x}}\Big) \in \Re^{n^y}$ and $\beta\Big(\tilde{\boldsymbol{x}}\Big) \in \Re^{n^y \times n^u}$ are also nonlinear functions. A vector $\tilde{\boldsymbol{y}} = [\boldsymbol{y}^T, \dot{\boldsymbol{y}}^T] \in \Re^{2n^y}$ will be used in the thesis for compact notation. Throughout this thesis, the nonlinear system (2.5) is considered.

2.2 Extraction of muscle synergies

Extraction of muscle synergies can be achieved by matrix factorization. Precisely, a set of N sample points of n^u-dimensional control signals $\{\boldsymbol{u}_i\}_{i=1}^N$ can be reconstructed by linear combinations of n^u vectors $\{\boldsymbol{w}_j\}_{j=1}^{n^u}$ without loss of information:

$$\boldsymbol{u}_i = \sum_{j=1}^{n^u} \Big(\boldsymbol{w}_j a_{ij}\Big) + \boldsymbol{w}_0 \tag{2.6}$$

where $\boldsymbol{w}_0 \in \Re^{n^u}$ is a constant vector. The vectors $\{\boldsymbol{w}_j\}_{j=1}^{n^u} \in \Re^{n^u}$ are the muscle synergies and the n^u scalars $\{a_{ij}\}_{j=1}^{n^u}$ are the corresponding synergy activations for reconstructing the sample point $\boldsymbol{u}_i$. The extracted muscle synergies are groups of muscle co-activations.

If all the control signals $\{\boldsymbol{u}_i\}_{i=1}^N$ lie close to a M-dimensional manifold of lower dimensionality than that of the original data space, the control signals can be approximated by linear combinations of fewer M muscle synergies

$$\begin{aligned} \boldsymbol{u}_i &\approx \sum_{j=1}^{M} \Big(\boldsymbol{w}_j a_{ij}\Big) + \boldsymbol{w}_0 \\ &= \mathbf{W}\boldsymbol{a}_i + \boldsymbol{w}_0 \end{aligned} \tag{2.7}$$

where $\mathbf{W} \in \Re^{n^u \times M}$ contains M of the n^u vectors in $\{\boldsymbol{w}_j\}_{j=1}^{n^u}$ and $\boldsymbol{a}_i \in \Re^M$ is the M-dimensional of synergy activations. The remaining $n^u - M$ synergies are stored in $\mathbf{W}_\perp \Re^{n^u \times (n^u - M)}$ such that $\mathbf{W} \cup \mathbf{W}_\perp = \{\boldsymbol{w}_j\}_{j=1}^{n^u}$.

Nonnegative matrix factorization (NMF) [105] is one of the widely used tools for extraction of muscle synergies in biological studies [25, 106]. Given nonnegative control signals $\{\boldsymbol{u}_i\}_{i=1}^N$, NMF extracts nonnegative muscle synergies $\{\boldsymbol{w}_j \geq \mathbf{0}\}_{j=1}^{n^u}$ and nonnegative synergy activations $\{a_{ij} \geq 0\}_{j=1}^{n^u}$ such that

$$\boldsymbol{u}_i = \sum_{j=1}^{n^u} \boldsymbol{w}_j a_{ij} \tag{2.8}$$

where the vector $\boldsymbol{w}_0$ in (2.7) becomes zeros in in this case. The nonnegative nature

provides direct insights how the actuators are coordinated in the extracted synergies.

Principal component analysis (PCA) is another widely used tool for extraction of muscle synergies [34, 35, 107]. It extracts muscle synergies as the orthogonal bases, which are known as the principal components, that the first principal component is colinear with the direction having the maximum variance of the data [108] such that

$$\boldsymbol{u}_i = \sum_{j=1}^{n^u} \boldsymbol{w}_j a_{ij} + \bar{\boldsymbol{u}} \tag{2.9}$$

where $\{\boldsymbol{w}_j\}_{j=1}^{n^u}$ are the principal components and the vector $\boldsymbol{w}_0$ in (2.7) becomes the mean value $\bar{\boldsymbol{u}}$ of $\{\boldsymbol{u}_i\}_{i=1}^{N}$. PCA is mainly employed in literature for the purpose of dimensionality reduction.

Other various matrix factorization techniques such as independent component analysis (ICA) [109, 110], factor analysis [111] (FA) have been used in the literature to extract muscle synergies. This thesis focuses on the functionality of dimensionality reduction, therefore the widely used tool NMF is employed in chapter 3 for additional purpose of understanding physical meaning of muscle coordination, and PCA in chapter 4 and chapter 5 for its algorithmic simplicity and the ease of implementation, respectively.

2.3 Control methods

Control of a robot refers to finding appropriate control inputs of the actuators in order to achieve a specific task. In this thesis, control techniques in optimal control theory and task space tracking control are applied for analysis of muscle synergies. This section gives a brief introduction about the optimal control theory and task space tracking control, and discuss the difficulty of using them in musculoskeletal robots.

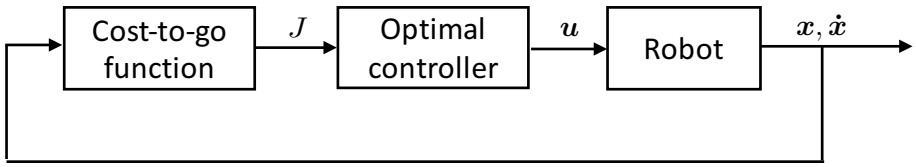


Figure 2-2: Schematic diagram of optimal control.

2.3.1 Optimal control theory

In optimal control theory, a control task is achieved by solving an optimization problem which in a cost-to-go function (or performance index) defining the task is minimized (or maximized), with given robot's dynamics model given. In the case of musculoskeletal robots, given the known dynamics model (2.5), an optimal controller (control law) is obtained by minimizing a cost-to-go function J in the following form

$$\begin{aligned} J\Big(\boldsymbol{x}(t_0)\Big) =& l_{final}\Big(\boldsymbol{x}(\bar{T}), \dot{\boldsymbol{x}}(\bar{T}), \boldsymbol{y}(\bar{T}), \dot{\boldsymbol{y}}(\bar{T})\Big) \\ &+ \int_{t_0}^{\bar{T}} l\Big(\boldsymbol{x}(t), \dot{\boldsymbol{x}}(t), \boldsymbol{y}(t), \dot{\boldsymbol{y}}(t), \boldsymbol{u}(t)\Big) dt \end{aligned} \tag{2.10}$$

where t_0 and $\bar{T}$ are the start and end time of the control task. l_{final} defines the cost at the end time and l defines the cost at intermediate time t.

Motion planning is carried out simultaneously when solving for optimal control $\boldsymbol{u}^*(t)$. $\boldsymbol{u}^*(t)$ is obtained by solving the optimization problem in backward time manner, such that the resulting trajectories of $\boldsymbol{x}(t)$, $\dot{\boldsymbol{x}}(t)$, $\boldsymbol{y}(t)$ and $\dot{\boldsymbol{y}}(t)$ are optimal with respect to J. Fig. 2-2 shows a schematic diagram of optimal control.

For example, consider a Linear Quadratic Regulator (LQR) setting:

$$\begin{aligned} \dot{\tilde{\boldsymbol{x}}}(t) =& \mathbf{A}\tilde{\boldsymbol{x}}(t) + \mathbf{B}\boldsymbol{u}(t) \\ J =& \frac{1}{2}\tilde{\boldsymbol{x}}^T(\bar{T})\mathbf{P}\tilde{\boldsymbol{x}}(\bar{T}) + \frac{1}{2}\int_{t_0}^{\bar{T}} \Big(\tilde{\boldsymbol{x}}^T(t)\mathbf{Q}\tilde{\boldsymbol{x}}(t) + \boldsymbol{u}^T(t)\mathbf{R}\boldsymbol{u}(t)\Big) \end{aligned} \tag{2.11}$$

where $\mathbf{A} = f\Big(\tilde{\boldsymbol{x}}\Big)\tilde{\boldsymbol{x}}$, $\mathbf{B} = g\Big(\tilde{\boldsymbol{x}}\Big)$ (compared with (2.5)). $\mathbf{P}$ and $\mathbf{Q}$ are symmetric, positive semidefinite matrices. $\mathbf{R}$ is symmetric, positive matrix. The optimal control

law for the LQR problem (2.11) is given by:

$$\begin{aligned} \boldsymbol{u}^*(t) =& -\mathbf{K}(t)\tilde{\boldsymbol{x}}(t) \\ \mathbf{K}(t) =& \mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}(t) \end{aligned} \tag{2.12}$$

where $\mathbf{S}(t)$ is the solution of the Riccati equation

$$-\dot{\mathbf{S}}(t) = \mathbf{S}(t)\mathbf{A} + \mathbf{A}^T\mathbf{S}(t) - \mathbf{S}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}(t) + \mathbf{Q}, \quad S(\bar{T}) = \mathbf{P} \tag{2.13}$$

which is solved in the backward time manner. When there is no time limit $\bar{T} = \infty$, the problem is called the infinite horizon problem and $\mathbf{S}(t)$ is the unique solution of the Algebraic Riccati equation $\mathbf{0} = \mathbf{S}(t)\mathbf{A} + \mathbf{A}^T\mathbf{S}(t) - \mathbf{S}(t)\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}(t) + \mathbf{Q}$. That is, the unique solution for $\dot{\mathbf{S}}(t) = \mathbf{0}$.

Solving for optimal control problems is generally difficult, because analytical solutions cannot be found in many cases, and the computation requirement dramatically increases with the duration of the time interval $[t_0, \bar{T}]$ and the dimensionality of the state space. There are several numerical approaches for solving optimal control problems. Indirect methods in which induced boundary-value problem is solved through iterations of integrations of the robot forward dynamics and so-called backward costate equations [112]. Direct methods where optimal control problems are discretized and transformed into nonlinear programming problems [113]. Dynamic programming (DP) is a well-known method that solves optimal control problems based on Bellman's principle of optimality that limits the number of potentially optimal control strategies, however, it still encounters the curse of dimensionality that hinders practical applications for high dimensional systems [114].

Utilizing muscle synergies, the dimensionality of the control space can be reduced. Using (2.7) to substitute the control variable $\boldsymbol{u}$ in the robot dynamics (2.5) and the cost-to-go function definition (2.10) to $\boldsymbol{a}$ which has lower dimensionality $M \leq n^u$. The resulting optimal control problem can then easier be solved in the reduced dimensionality.

2.3.2 Task-space tracking control

The control task of task-space tracking control is to follow a pre-defined task space trajectory, in contrast to the optimal control problem where a task space trajectory is implicitly computed during solution solving. The control problem is to find a control law to track a pre-defined task space trajectory $\tilde{\boldsymbol{y}}^*(t) = \Big[\boldsymbol{y}^*(t), \dot{\boldsymbol{y}}^*(t)\Big]$ in time interval $t \in [t_0, \bar{T}]$ such that the tracking position error $\boldsymbol{e}(t) = \boldsymbol{y}(t) - \boldsymbol{y}^*$ and the velocity error $\dot{\boldsymbol{e}}(t) = \dot{\boldsymbol{y}}(t) - \dot{\boldsymbol{y}}^*$ are limited by small values $\boldsymbol{\epsilon}_1$ and $\boldsymbol{\epsilon}_2$:

$$\forall t \geq 0, |\boldsymbol{e}(t)| \leq \boldsymbol{\epsilon}_1, |\dot{\boldsymbol{e}}(t)| \leq \boldsymbol{\epsilon}_2, \boldsymbol{\epsilon}_1 > 0, \boldsymbol{\epsilon}_2 > 0. \tag{2.14}$$

where $|\cdot|$ is an entry-wise operator returning absolute values.

Consider the musculoskeletal robot (2.5). In order to track a given desired task space trajectory $\tilde{\boldsymbol{y}}^*(t), t \in [t_0, \bar{T}]$ with corresponding desired acceleration $\ddot{\boldsymbol{y}}^*(t)$, the control input at time t must satisfy the task space dynamics in the last equation in (2.5). The general solution can be expressed as:

$$\boldsymbol{u}(t) = \beta^\dagger\Big(\tilde{\boldsymbol{x}}(t)\Big)\Big[\ddot{\boldsymbol{y}}^*(t) - \alpha\Big(\tilde{\boldsymbol{x}}(t)\Big)\Big] + \Big[\mathbf{I} - \beta^\dagger\Big(\tilde{\boldsymbol{x}}(t)\Big)\beta\Big(\tilde{\boldsymbol{x}}(t)\Big)\Big]\boldsymbol{\xi} \tag{2.15}$$

where $\beta^\dagger\Big(\tilde{\boldsymbol{x}}(t)\Big)$ is the Moore-Penrose inverse of $\beta\Big(\tilde{\boldsymbol{x}}(t)\Big)$, $\mathbf{I}$ is the identity matrix. $\boldsymbol{\xi}$ is an arbitrary vector. In ideal case where the robot can be exactly modeled by (2.5) and the nonlinear functions f, g, h, α and β are exactly correct, the control law (2.15) can achieve tracking the desired trajectory $\tilde{\boldsymbol{y}}^*$. In reality, it is impossible to obtain exact model of the robot. One common usual approach in engineering is to add a feedback control term $\boldsymbol{u}_{ff}(t)$ to (2.15) such that the error dynamics [83, 115, 116]:

$$\ddot{\boldsymbol{e}} + \mathbf{K}_v\dot{\boldsymbol{e}} + \mathbf{K}_p\boldsymbol{e} = 0 \tag{2.16}$$

is stable, i.e. $\boldsymbol{e} \to \mathbf{0}$ as $t \to 0$, where $\mathbf{K}_v$ and $\mathbf{K}_p$ are the control gain matrices.

The resulting tracking control law $\boldsymbol{u}(t)$ consists of three components:

$$\boldsymbol{u}(t) = \boldsymbol{u}_{ff}(t) + \boldsymbol{u}_{fb}(t) + \boldsymbol{u}_{null}(t) \tag{2.17}$$

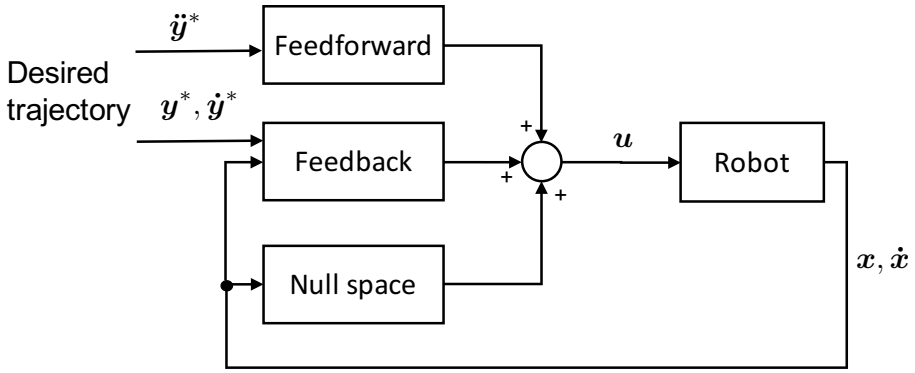


Figure 2-3: Schematic diagram of task space tracking control.

where $\boldsymbol{u}_{ff}(t) = \beta^{\dagger}\left(\tilde{\boldsymbol{x}}(t)\right)\left[\ddot{\boldsymbol{y}}^*(t) - \alpha\left(\tilde{\boldsymbol{x}}(t)\right)\right]$ is the feedforward control term (the first term in (2.15)), which is responsible for achieving desired acceleration $\ddot{\boldsymbol{y}}^*$ in the task space. This feedforward control term can be computed either from given analytical model of the robot dynamics (functions $\alpha\left(\tilde{\boldsymbol{x}}(t)\right)$ and $\beta\left(\tilde{\boldsymbol{x}}(t)\right)$), or estimated from data. In adaptive control approach [77], this term is updated online from new measurement data to adapt environmental changes. $\boldsymbol{u}_{null}(t) = \left[\mathbf{I} - \beta^{\dagger}\left(\tilde{\boldsymbol{x}}(t)\right)\beta\left(\tilde{\boldsymbol{x}}(t)\right)\right]\boldsymbol{u}_0$, where $\boldsymbol{\xi} = \boldsymbol{u}_0$, is the null space control term (the second term (2.15)), which acts in the null space of $\beta\left(\tilde{\boldsymbol{x}}\right)$ such that the tracking performance in the task space is not interfered. The null space control term is used for joint stabilization and achieving secondary task goal that is defined by $\boldsymbol{u}_0$. It has been demonstrated that the joints can be "pulled" to desired joint angles [116]. $\boldsymbol{u}_{fb}$ is the feedback control term responsible for compensating modeling error to achieve asymptotic tracking in task space. In chapter 5, the feedback control term is designed based on sliding mode control [77] approach. Fig. 2-3 shows a schematic diagram of task space tracking control.

Utilizing muscle synergies (2.7), the computation of the tracking control (2.17) can be reduced by decreasing the control dimensionality similar to the case in optimal control. The computation of the tracking control (2.17) involves computation of the pseudo inverse of $\beta\left(\tilde{\boldsymbol{x}}\right) \in \Re^{n^y \times n^u}$ (also inversion of $g\left(\tilde{\boldsymbol{x}}\right) \in \Re^{n^y \times n^u}$ may be needed for the null space control term). The computation cost of the pseudo inverse can be reduce by reducing the matrix $\beta\left(\tilde{\boldsymbol{x}}\right) \in \Re^{n^y \times n^u}$ to a $n^y \times M$ matrix.

Chapter 3

Analysis of muscle synergies and its utilization for generation of optimal movements

This chapter verifies the feasibility of utilizing muscle synergies to reduce the control dimensionality in controlling a musculoskeletal robot. One of the main difficulties in controlling musculoskeletal robots is to determine the appropriate control inputs to the many actuators. It has been suggested that human reduces control dimensionality by coordinating groups of muscle co-activations called muscle synergies, instead of controlling muscles independently. In this chapter, the feasibility is investigated in simulation experiments, where control performance is compared between utilizing different sets of muscle synergies, which are extracted from sequences of control signals having particular inherent characteristics that actuate the robot's end effector to produce omnidirectional movements in the task space. In the experiments, a human-like robotic arm utilized a set of muscle synergies to move the end effector to a set of targets spreading in the task space of the robot, where the control signals were determined by minimizing the final distance from a target and the minimum control effort. Among five sets of muscle synergies being investigated, it was found the robot could be controlled utilizing the following two sets of muscles synergies: 1) The goal-related synergies, which were extracted from a data sample of optimized movements

that had minimum distances from targets at the final time step and minimum total control effort spent, where the control inputs to actuate the robot was determined by an algorithm in the optimal control theory. 2) The energy-efficient synergies, which were extracted from a data sample of optimized movements according to a fitness function defined as the ratio between kinetic energy and the movement and control, where genetic algorithm was employed for the optimization. In the above two cases, it was found that the control dimensionality could be reduced from 10 to 5 in reaching a set of targets ranging 30cm to 40cm away from the initial end effector positions. The success of utilizing the goal-related synergies implies muscle synergies extracted from control signals that are optimized to achieve specific task goals can be utilized to reduce control dimensionality in achieving the same task goals. The success of utilizing the energy-efficient synergies implies that goal-directed tasks could be achieved by muscle synergies extracted from optimized control signals with respect to energy efficiency, regardless of whether task goals are specified in the optimization.

3.1 Introduction

Musculoskeletal robots are expected to have the ability to behave like biological creatures because of their similar mechanical structures. The control of such complex structure is also difficult.

Within the field of human motor control research, the hypothesis of muscle synergies [117–119] has been proposed as a solution to the degree-of-freedom problem. Among many interpretations of muscle synergies, one suggests that many muscles are not controlled individually, but a few groups of muscles with specific activation patterns, namely muscle synergies, are coordinated. Control can then be simplified by coordinating fewer control variables in terms of muscle synergies.

Several works have shown that human-like goal-directed movements can be predicted by the optimal control theory [68, 120, 121]. But the curse of dimensionality [122] in solving the optimal control problem is still one of the main difficulties to be overcome in engineering.

In this chapter, application of muscle synergies in controlling musculoskeletal robotic system by optimal control is studied. Special attention is paid to the following problems: 1) Can muscle synergies facilitate the solving of optimal control problem? 2) What properties should muscle synergies have in order to achieve tasks? 3) How should muscle synergies be utilized better?

In addition to the control complexity reduction by utilizing muscle synergies, the main contributions of this research are the understanding of inherent properties of muscle synergies and the ways to obtain the muscle synergies. The performance is analyzed by utilizing two types of muscle synergies, namely *goal-related synergies* and *goal-unrelated synergies*. The former ones are extracted from solutions optimized according to the cost-to-goal function specified as task goals, whereas the latter ones are extracted from solutions optimized according to different fitness criteria having different meanings of "movement fitness" instead of explicit task goals. Moreover, since the *goal-unrelated synergies* are obtained according to different optimization criteria, successful achievement of some particular tasks utilizing the *goal-unrelated synergies* would imply that muscle synergies can be extracted by other methods (e.g. GA), rather than from solutions of the optimal control problem.

Similar studies on muscle synergies properties can be found in [96], where the muscle synergies were obtained from optimal control solutions only. They studied the time-varying synergies which represent the spatiotemporal actuation pattern. In contrast, time-invariant synergies are studied in this chapter. A time-invariant synergy represents a spatial muscle co-activation pattern. In order to achieve a novel task, time-invariant muscle synergies might be better because they are task-independent. To achieve a (novel) task, the time-invariant synergies are coordinated by the corresponding time-varying synergy activations, which possess task-related information and are determined by a controller in use.

3.2 Optimal control utilizing muscle synergies

This chapter considers a control task of a musculoskeletal robot end effector. The control task is a reaching task that the robot is required to move the end-effector to a target position $\boldsymbol{y}*$ and with desired velocity $\dot{\boldsymbol{y}}*$ from joint configuration $\boldsymbol{x}(t_0)$ from starting time t_0. The reaching task is achieved by solving the following optimal control problem for the musculoskeletal system (2.5), where the optimal muscle activation $\boldsymbol{u}^*(t)$ is sought to minimize the cost-to-go function $\tilde{J}\left(\boldsymbol{x}(t_0)\right)$:

$$\begin{aligned} \ddot{\boldsymbol{x}}(t) &= f\Big(\tilde{\boldsymbol{x}}(t)\Big) + g\Big(\tilde{\boldsymbol{x}}(t)\Big)\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= h\left(\boldsymbol{x}(t)\right) \\ \dot{\boldsymbol{y}}(t) &= \frac{\partial}{\partial t}\Big(h\left(\boldsymbol{x}(t)\right)\Big) \\ \tilde{J}\left(\boldsymbol{x}(t_0)\right) &= l_{final}\left(\tilde{\boldsymbol{y}}(\bar{T}), \tilde{\boldsymbol{y}}*\right) + \int_{t_0}^{\bar{T}} l\left(\boldsymbol{u}(t)\right) dt \end{aligned} \tag{3.1}$$

where $\boldsymbol{x}$, $\dot{\boldsymbol{x}}$ and $\ddot{\boldsymbol{x}}$ are joint angles, joint velocities and joint accelerations, respectively. $\tilde{\boldsymbol{x}} = [\boldsymbol{x}^T, \dot{\boldsymbol{x}}^T]$. $\boldsymbol{u}$ specifies the muscle activations to the muscles, where the muscle activations are nonnegative and bounded by an upper limit $\boldsymbol{u}^{ub}$ such that $\mathbf{0} \leq \boldsymbol{u} \leq \boldsymbol{u}^{ub}$. $\tilde{\boldsymbol{y}} = [\boldsymbol{y}^T, \dot{\boldsymbol{y}}^T]$. $\tilde{\boldsymbol{y}}*$ is the target position and velocity. t_0, $\bar{T}$, l_{final} and l denote start time, final time, final state cost and instantaneous cost, respectively. The functions f, g and h describe the robot dynamics and observer, respectively. With the aid of muscle synergies, the control dimensionality can be reduced. The nonlinear optimal control framework iterative Linear Quadratic Gaussian (iLQG) [123] was adopted to solve for optimal solutions.

Because biological muscles can only provide contraction forces and have non-negative control signals, all the actuator inputs $\boldsymbol{u}(t)$, the synergies $\mathbf{W}$ and synergy activations $\boldsymbol{a}(t)$ are constrained to be non-negative for better understanding of the muscles (actuators) activations in solutions and mimic biological control system. The nonnegative muscle synergies are extracted by using nonnegative matrix factorization (NMF). The extraction will be explained later in section 3.3. A muscle synergy extracted by NMF represents a synchronous activation pattern of a group of actuators

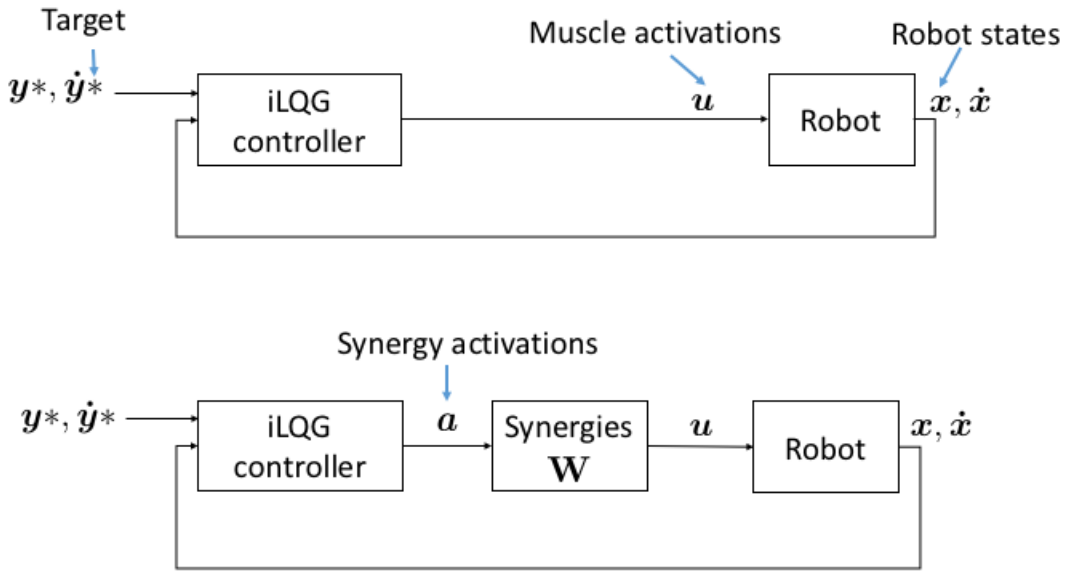


Figure 3-1: Schematic diagram of the control method utilizing muscle synergies. When the robot is controlled in the original dimensionality (the upper figure), the controller computes muscle activations that actuate and move the robot to the target end-effector position and velocity based on the feedback of the robot states. When synergies are used (the lower figure), the controller computes the synergy activations, which have lower dimensionality than the muscle activations. The synergies transform the synergy activations to the muscle activations, which actuate and move the robot to the target end-effector position and velocity.

(muscles). Control signals of n actuators can be approximated as linear combinations of M ($M \leq n$) muscle synergies:

$$\boldsymbol{u}(t) \approx \sum_{j=1}^{M} \boldsymbol{w}_j a_j(t) = \mathbf{W}\boldsymbol{a}(t) \tag{3.2}$$

where $\boldsymbol{u}(t) \in \Re^n$ is the actuator input at time t, $\boldsymbol{w}_j \in \Re^{n^u}$ and $a_j(t) \in \Re^1$ are synergy j and the corresponding synergy activation, respectively. $\mathbf{W}$ is an $n^u \times M$ matrix that collects all M synergies, and is constant for all time t.

Given a set of synergies $\mathbf{W} \in \Re^{n \times M}$ where $M < n$, the control space is transformed into one with lower dimensionality as shown in equation (3.2). Then the optimal

control problem can be solved in lower dimensionality with respect to $\boldsymbol{a} \in \Re^M$:

$$\begin{aligned} \ddot{\boldsymbol{x}}(t) &= f\Big(\tilde{\boldsymbol{x}}(t)\Big) + g\Big(\tilde{\boldsymbol{x}}(t)\Big)\mathbf{W}\boldsymbol{a}(t) \\ \boldsymbol{y}(t) &= h\Big(\boldsymbol{x}(t)\Big) \\ J\left(\boldsymbol{x}(t_0)\right) &= \tilde{l}\Big(\tilde{\boldsymbol{y}}(\bar{T}), \tilde{\boldsymbol{y}}*\Big) + \int_0^{\bar{T}} l\Big(\mathbf{W}\boldsymbol{a}(t)\Big)dt \end{aligned} \tag{3.3}$$

where the optimal synergy activation $\boldsymbol{a}^*(t)$ is sought such that the cost-to-go function $J(\boldsymbol{x}(t_0))$ is minimized.

Fig. 3-1 depicts the schematic diagrams controlled by the iLQG controller in original dimensionality and in reduced control dimensionality via muscle synergies.

3.3 Muscle synergies with different properties

This chapter analyzes different synergies extracted from movements that are generated in different ways. Three types of muscle synergies, namely *goal-related synergies*, *goal-unrelated synergies* and *random synergies* will be analyzed in this chapter. The ways generating different movements to extract the three types of muscle synergies are described.

3.3.1 Extracting muscle synergies

NMF is adopted to extract synergies, such that the root-mean squared residual $(\|\mathbf{U}^0 - \mathbf{U}^M\|_f/(N^u))^{1/2}$ between the original signals $\mathbf{U}^0$ and the reconstructed signals $\mathbf{U}^M = \mathbf{W}^M\mathbf{A}^M$ is minimized, where N^u is the number of elements in the matrix $\mathbf{U}^0$ (or $\mathbf{U}^M$), $\mathbf{W}^M \in \Re^{n\times M}$ is the set of M synergies extracted, $\mathbf{A}^M \in \Re^{M\times(N^u/n)}$ is the corresponding synergy activations. Extraction is performed by function *nnmf* in the MATLAB statistic toolbox. For example, a synergies set $\mathbf{W}^M \in \Re^{n\times M}$ can be extracted from r movement, by applying NMF to a control signal matrix $\mathbf{U}^0 = [\boldsymbol{u}_1^0, \boldsymbol{u}_2^0, ..., \boldsymbol{u}_r^0]$ concatenated from all r movements which are gathered based on a certain criterion, where $\boldsymbol{u}_j^0, j = 1, ..., r$ is the series of control signals of movement

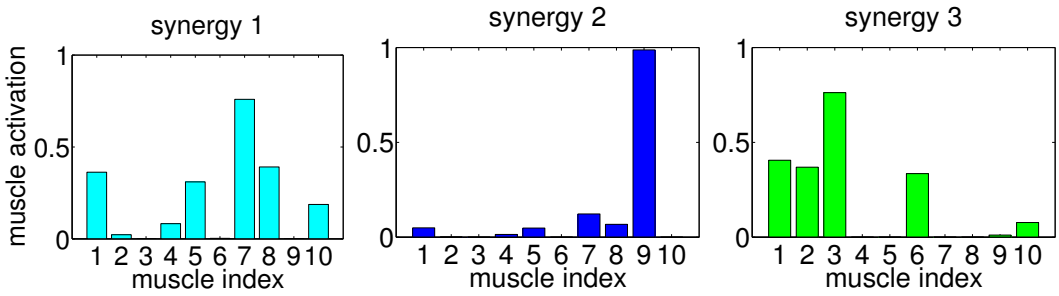


Figure 3-2: A set of 3 synergies extracted by NMF ($M = 3$). Each synergy is time invariant, specifying synchronous activations of all the 10 muscles.

with index j. All synergies extracted by NMF are then normalized as follows

$$w_{ij} \leftarrow \frac{w_{ij}}{\sqrt{\sum_{k=1}^{n}(w_{kj})^2}} \tag{3.4}$$

where $\mathbf{W} = \{\boldsymbol{w}_j, j = 1, ..., M\}$, and $\boldsymbol{w}_j = \{w_{ij}, i = 1, ..., n\}$. In the following sections it will be described in detail how to gather movements and obtain different synergies by using different criteria. Fig. 3-2 depicts an example of a set of synergies consisting of 3 synergies.

3.3.2 Type 1: goal-related synergies

goal-related synergies are extracted from the optimal control solutions of (3.1), that is, the optimal solutions without utilizing synergies. It is named as "goal-related" because they are extracted from the optimal solutions of (3.1), having the property of achieving a goal that is explicitly specified in the cost-to-goal function. (e.g. reaching targets in Cartesian coordinates). It is reasonable that the same goals can be achieved by utilizing *goal-related synergies*. The *goal-related synergies* are denoted as $\mathbf{W}_*^M \in \Re^{n \times M}$ where n and M are the dimension of control signals and the number of synergies extracted, respectively.

3.3.3 Type 2: goal-unrelated synergies

goal-unrelated synergies are extracted from control signals of movements that are generated by optimizing movements according to specific movement fitness criteria. The movement fitness is not related to any explicit task goal such as targets in Cartesian coordinates. Therefore, the resulting movements are different from the goal-directed movements generated by solving the optimal control problem. In this chapter *goal-unrelated synergies* are extracted from some "good" movements that are generated by using genetic algorithm (GA) with three different movement fitness criteria, namely *energy-efficient*, *curvature* and *jerk*. Details of generating such movements are as follows.

A movement $\xi = \{\boldsymbol{x}_t, \boldsymbol{u}_t\}, t = 1, ..., \bar{T}$ is generated from a given configuration $\boldsymbol{x}_0$ and final time $\bar{T}$ using the arm dynamics described in the Appendix. The activation of a muscle (actuator) i at time t is defined by a Gaussian function with 3 parameters controlling the amplitude A_i, the width σ_i and the center c_i of the Gaussian:

$$u_i(t) = A_i \exp\left((t - c_i)^2/2\sigma_i\right), t = 1, ..., \bar{T}. \tag{3.5}$$

A "good" movement ξ_q is generated by optimizing the activation parameters $\Xi_q = [A_i, \sigma_i, c_i]_q, i = 1, ..., 10$ of each muscle such that the value of the movement according to the following movement fitness functions $V_q, q = \{en, cur, jerk\}$ are minimized:

$$\begin{aligned} V_{en} &= \frac{\text{force input}}{\text{K.E}} = \frac{\sum_{t=1}^{\bar{T}} \left(\sum_{i=1}^{10} \hat{f}^i\right)}{\sum_{t=1}^{\bar{T}} \left(\dot{\theta}_t^T H(\theta_t) \dot{\theta}_t\right)} \\ V_{cur} &= \text{maximum curvature} = \max_{t \in [2,...,\bar{T}]} |\angle_t| \\ V_{jerk} &= \text{maximum jerk} = \max_{t \in [3,...,\bar{T}]} \left(\dddot{p}_{x_t}^2 + \dddot{p}_{y_t}^2\right) \end{aligned} \tag{3.6}$$

where V_{en}, V_{cur} and V_{jerk} relate to input-output efficiency, smoothness of trajectory in the Cartesian plane, and jerk of end-effector trajectory in the Cartesian plane, respectively. $\angle_t$ is the angle between successive end-effector velocities: $\angle_t = |\cos^{-1}(< \hat{\boldsymbol{v}}(t-1), \hat{\boldsymbol{v}}(t) >)|$ where $< \hat{\boldsymbol{v}}(t-1), \hat{\boldsymbol{v}}(t) >$ denotes the dot product of suc-

cessive normalized end-effector velocities with $\hat{\boldsymbol{v}} = \boldsymbol{v}/(||\boldsymbol{v}||_2)$.

The reciprocal of V_{en} measures the ratio between the total kinetic energy output and the total force input to the robot. Therefore, a smaller value of V_{en} implies a better energy efficiency. V_{cur} measures the smoothness by the maximum change of trajectory tangent. V_{jerk} is the modified version of measuring trajectory smoothness defined in [124], where $\dddot{p}$ denotes the rate of change of acceleration.

Each movement ξ_q is optimized with given random final time $\bar{T}$ and random resting (zero velocities) configuration $\boldsymbol{x}_0$ with respect to each movement fitness criterion V_q by adopting GA to search for values of $\Xi_q = [A_i, \sigma_i, c_i]_q, i = 1, ..., 10$, such that V_q is minimized. The GA optimization is performed by using function *ga* in MATLAB. For each movement, the population size of Ξ_q is set to 20 and then 5 generations are carried out before the termination of evolution.

$\mathbf{W}_{en}^M$, $\mathbf{W}_{cur}^M$ and $\mathbf{W}_{jerk}^M$ are used to denote the *energy-efficient synergies*, *curvature synergies*, the *jerk synergies*, respectively, where M is the number of synergies extracted.

3.3.4 Type 3: Random synergies

In order to verify the necessity of the inherent properties in muscle synergies, Non-negative *random synergies* are constructed for comparison. The *random synergies* that are drawn from identical distribution and normalized by using (3.4). $\hat{\mathbf{W}}^M$ denotes the *random synergies*.

3.4 Performance Analysis

Reaching task is a common goal-directed movement in human motion. This chapter analyzes reaching movements generated by utilizing muscle synergies, and put focuses on the following issues:

1. Whether muscle synergies can facilitate the solving of the optimal control problem?

2. What properties should muscle synergies have?

The reaching movements starting from a joint configuration $\boldsymbol{x}$ at time $t = 0$ are achieved by finding an optimal solution that minimizes the cost-to-go function:

$$\begin{aligned} J\left(\boldsymbol{x}(t_0)\right) = & \frac{1}{2}\left(\tilde{\boldsymbol{y}}(\bar{T}) - \tilde{\boldsymbol{y}}*\right)^T \mathbf{P}\left(\tilde{\boldsymbol{y}}(\bar{T}) - \tilde{\boldsymbol{y}}*\right) \\ & + \frac{1}{2}\int_{t_0}^{\bar{T}} \left(\boldsymbol{u}(t)\right)^T \mathbf{R}\left(\boldsymbol{u}(t)\right) \end{aligned} \tag{3.7}$$

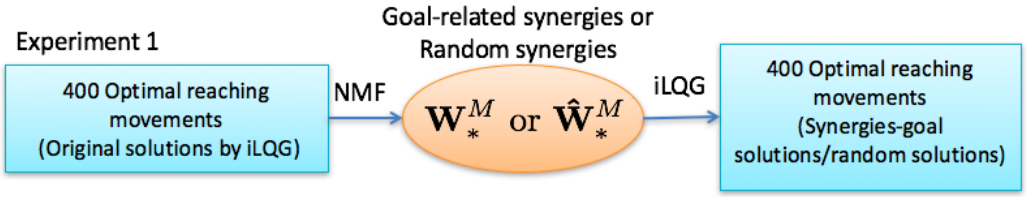
where $\boldsymbol{x}$ and $\boldsymbol{u}$ are the robot joints and muscle activations. $\tilde{\boldsymbol{y}}$ is a vector containing position and the velocity of the end-effector in the Cartesian coordinate plane and $\tilde{\boldsymbol{y}}*$ denotes the target position and velocity. The movement time $\bar{T}$ was determined by Fitts's law $\bar{T} = a + b\log_2(1 + 2d/w)$ to mimic human motion behavior, where d is the distance from the target and w is the tolerance. The values of parameters $\mathbf{P} = \text{diag}(10000, 10000, 1000, 1000)$, $\mathbf{R} = \mathbf{I}$, $a = 0$, $b = 0.08$, $w = 0.02(\approx 2cm)$ are used. With this setting the cost-to-go value J approximately equals the Euclidean distance d from desired targets in centimeter ($J = 1$ implies $d \approx 1cm$ in our case).

For simplicity, the following are defined:

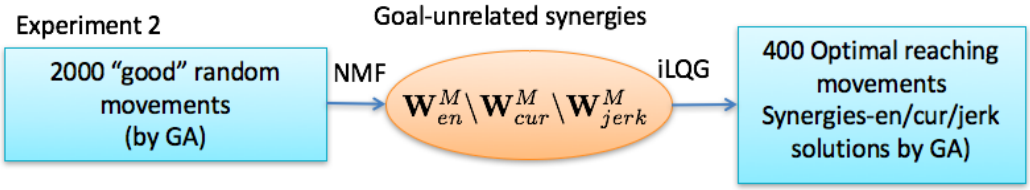
- *Original solutions*: Solutions without utilizing synergies by solving (3.1) directly.
- *Synergies-goal solutions*: Solutions utilizing *goal-related synergies* by solving (3.3).
- *Synergies-en/cur/jerk solutions*: Solutions utilizing *goal-unrelated synergies* by solving (3.3), in which the synergies are extracted from movements generated by GA according to movement fitness functions V_{en}, V_{cur} or V_{jerk}.
- *Random solutions*: Solutions utilizing *random synergies* by solving (3.3).

3.4.1 Experiment 1: Feasibility of muscle synergies approach

The objective of this experiment is to verify the feasibility of muscle synergies approach in solving the optimal control problem. *goal-related synergies* are utilized in



(a) Experiment 1: The *original solutions* are solved firstly, followed by extraction of *achieve-goal synergies* $\mathbf{W}_*^M$ using NMF. Then $\mathbf{W}_*^M$ are utilized for solving *synergies-goal solutions*. The *random solutions* are obtained by utilizing *random synergies* $\hat{\mathbf{W}}^M$.



(b) Experiment 2: Each *goal-unrelated synergies* $\mathbf{W}_{en}^M$, $\mathbf{W}_{cur}^M$ and $\mathbf{W}_{jerk}^M$ are extracted from a set of 2000 "good" random movements generated by GA according to criteria V_{en}, V_{cur} and V_{jerk} respectively. Then the *synergies-en/cur/jerk solutions* are obtained by utilizing the *goal-unrelated synergies* $\mathbf{W}_{en}^M$, $\mathbf{W}_{cur}^M$ and $\mathbf{W}_{jerk}^M$.

Figure 3-3: Summary of (a) Experiment 1 and (b) Experiment 2. All reaching movements were carried out using the same 400 pairs of initial and target positions $\{\boldsymbol{x}_0, \boldsymbol{y}*\}$.

this experiment. Because it is reasonable that *goal-related synergies* have properties to achieve the same goals. It would be said that the muscle synergies approach is feasible if its (*synergies-goal solutions*)

- success rate of reaching target has similar level, and
- computation expense is reduced

when compared with *original solutions*. In the next experiment, the properties of muscle synergies will be explored.

The experiment procedure is summarized in Fig. 3-3. In this experiment, 400 pairs of initial and target position of the end effector $\{\boldsymbol{x}_0, \boldsymbol{y}*\}$ were uniformly distributed over the workspace such that there were 100 pairs in each of the four target distance $d = \|\boldsymbol{y}_0 - \boldsymbol{y}*\|_2^2$ ranges (reaching distance): [0-10]cm, [10-20]cm, [20-30]cm and [30-40]cm. i) Firstly, the 400 *original solutions* were simulated. Then *goal-related synergies* $\mathbf{W}_*^M$ were extracted from the control signals in the *original solutions* by NMF as described in section 3.3. Eight sets of synergies $\mathbf{W}_*^M$ were extracted where

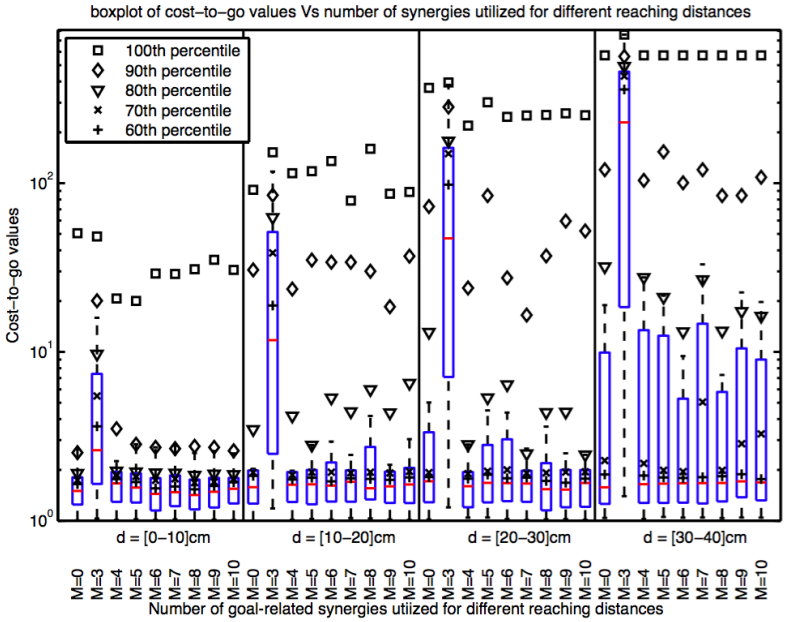
$M = 3, ..., 10$. ii) Then, for each $M = 3, ..., 10$, the 400 *synergies-goal solutions* were simulated utilizing $\mathbf{W}_*^M$ using the same 400 pairs of $\{\boldsymbol{x}_0, \boldsymbol{y}*\}$. iii) Finally, the 400 *random solutions* were simulated for verifying the necessity of the inherent characteristics in muscle synergies.

The optimal control problem solutions were solved by adopting the iLQG [123]. Iteration of iLQG was terminated if improvement of cost-to-go value is smaller than 10^{-3} or if the cost-to-go value is smaller than 1 (i.e. distance from target$< 1cm$).

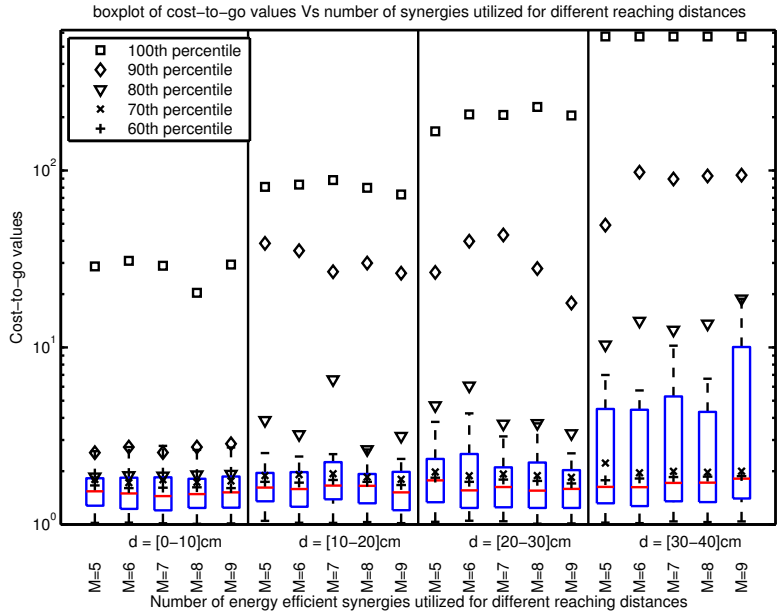
The box-plots of cost-to-go values ($\log_{10}$ scaled) and total computation time spent of utilizing synergies $\mathbf{W}_*^M$ for different reaching distances are depicted in Figs. 3-4a and 3-6a, respectively. Because a smaller value of cost-to-go value means a closer distance from the desired target, a lower position of the percentiles markers implies a better success rate. In Fig. 3-4a it is observed that when more than 3 synergies are utilized ($M > 3$), the cost-to-go values are not much different from the original solutions ($M = 0$) for all the four reaching distance ranges. On the other hand, it is observed in Fig. 3-6a that the computation expense is reduced as expected. The computation time spent is less than the original solutions and lesser time is spent when fewer synergies are utilized. In particular, from careful inspection at the data's percentile markers it is observed that the success rate is *better* in the *synergies-goal solutions*, especially for reaching further targets, since the percentiles markers has lower positions in the box-plots.

Fig. 3-5 depicts the performance of utilizing *random synergies* $\hat{\mathbf{W}}^M$. In contrast to the *synergies-goal solutions*, it is obvious that targets could not be reached in *random solutions*. This implies that synergies must possess inherent properties in some sense. For better illustration, the end-effector trajectories of a movement are depicted in Fig. 3-7. It is obvious that the resulting trajectories are closed to the *original solutions* (dash line in the figure) when more synergies are utilized. Moreover, the success rate is better when synergies are utilized in this reaching task because the resulting cost-to-go values are lower.

Fig. 3-8 depicts the induced accelerations by synergies of the trajectory depicted in Fig. 3-7 (M=5). The induced acceleration vector $\tilde{\boldsymbol{a}}^i(k)$ at time step k is computed



(a) Cost-to-go values of goal-related synergies



(b) Cost-to-go values of energy-efficient synergies

Figure 3-4: Performances of utilizing different numbers of synergies for different reaching distances. The cost-to-go values of utilizing goal-related synergies $\mathbf{W}^M$ and energy-efficient synergies $\mathbf{W}_{en}^M$ are depicted in (a) and (b), respectively. Each column represents the results of the 100 movements. "M=0" and "M=[3-10]" refer to the original solutions utilizing no synergies, and the number of synergies utilized. The two edges and the horizontal line in each box indicate the 25th, 75th and the median (50th) percentile of the data. The 60th, 70th, 80th, 90th and 100th percentiles are indicated by markers '+', '×', '▽', '⋄' and 'box', respectively. It can be observed that utilizing more than 3 ($M > 3$) goal-related synergies or energy-efficient synergies has a similar success rate compared with the *original solutions*.

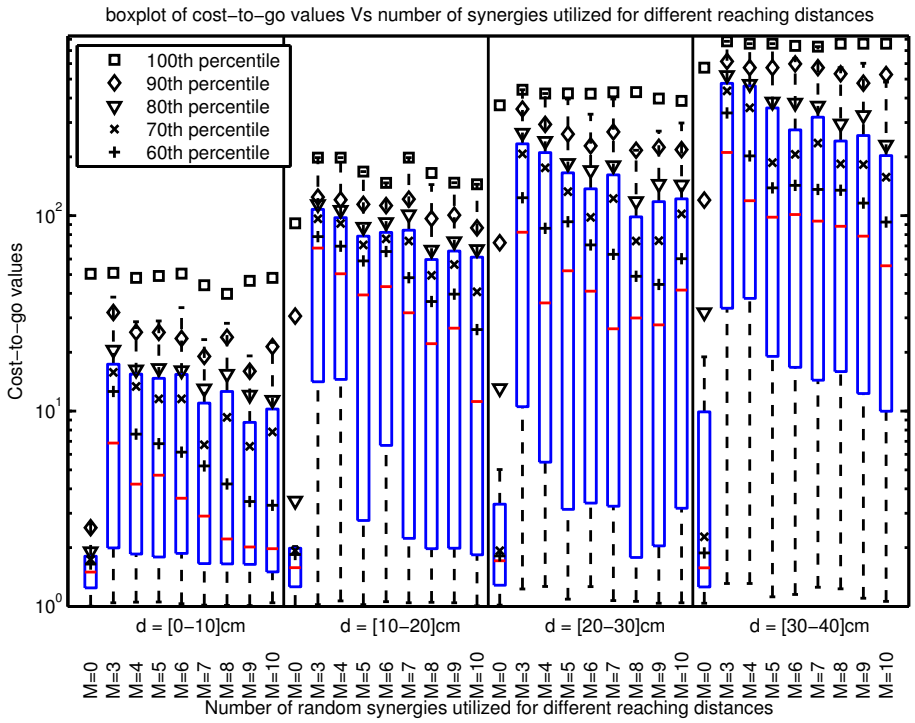
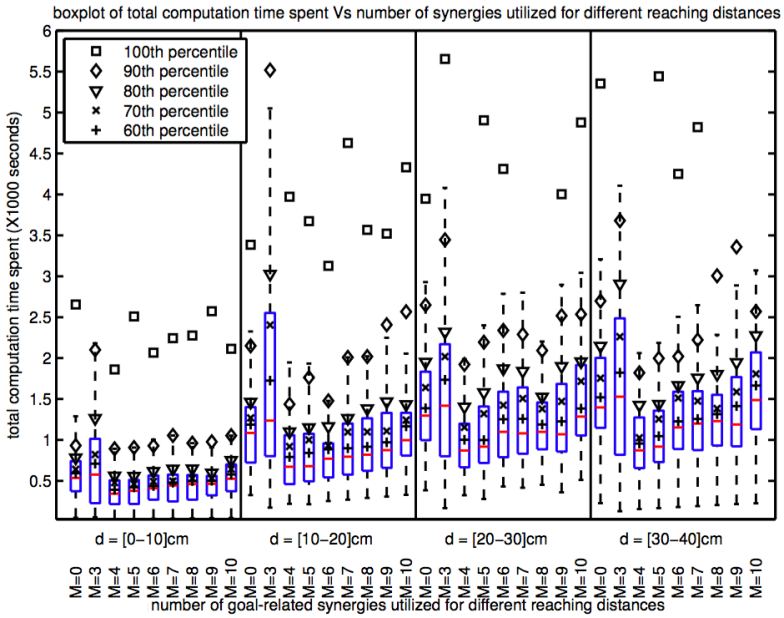


Figure 3-5: Performance of utilizing random synergies $\hat{\mathbf{W}}^M$. It can be observed that goals could not be achieved.

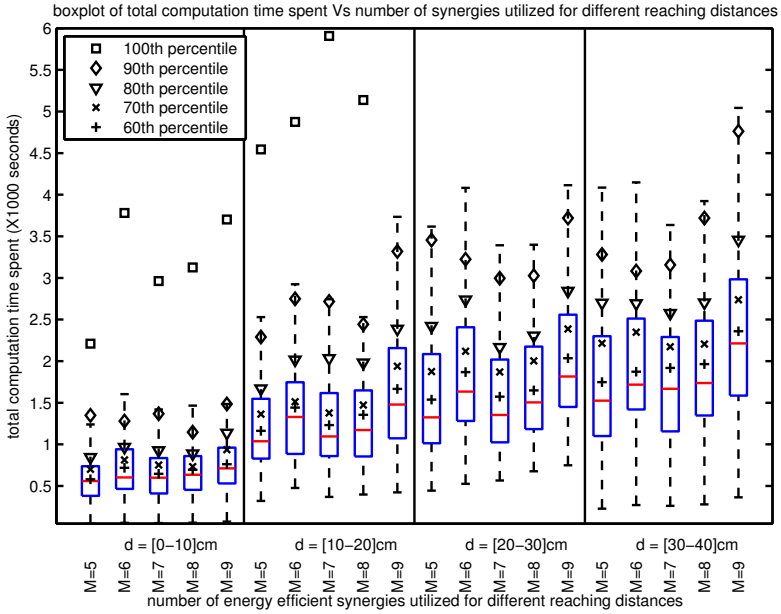
from the difference of the end-effector velocities in the task space: $\tilde{\boldsymbol{a}}^i(k) = \tilde{\boldsymbol{v}}_i(k+1) - \boldsymbol{v}(k)$, where $\tilde{\boldsymbol{v}}_i(k+1)$ is the task space velocities after applying only a synergy $\boldsymbol{w}_i$ with maximum activation a_i^{max}, i.e. $\boldsymbol{u}(t) = \boldsymbol{w}_i a_i^{max}$, at current state $\boldsymbol{x}_k$. It can be observed that the directions of the induced accelerations are quite constant, and spread towards different directions to span the task space. This activation pattern can be interpreted as follows: At the beginning, synergy 3 brings the end-effector towards the target and synergy 1 acts in the opposite direction to control the speed. Near the end of the trajectory, the acting of synergy 3 decreases and the other four synergies increase in order to stop at the target.

3.4.2 Experiment 2: Synergies with different properties

In the previous experiment, it was verified that utilizing *goal-related synergies* can solve the optimal control problem. In this experiment, the focus is extended to investigate what properties of muscle synergies should have in order to solve the



(a) Total computation time of goal-related synergies



(b) Total computation time of energy-efficient synergies

Figure 3-6: Performances of utilizing different numbers of synergies for different reaching distances. The total computation times spent of utilizing goal-related synergies $\mathbf{W}^M$ and energy-efficient synergies $\mathbf{W}_{en}^M$ are depicted in (a) and (b), respectively. Each column represents the results of the 100 movements. "M=0" and "M=[3-10]" refer to the original solutions utilizing no synergies, and the number of synergies utilized. The two edges and the horizontal line in each box indicate the 25th, 75th and the median (50th) percentile of the data. The 60th, 70th, 80th, 90th and 100th percentiles are indicated by markers '+', '×', '▽', '◇' and 'box', respectively. It can be observed that utilizing more than 3 ($M > 3$) goal-related synergies or energy-efficient synergies could reduce computation expense.

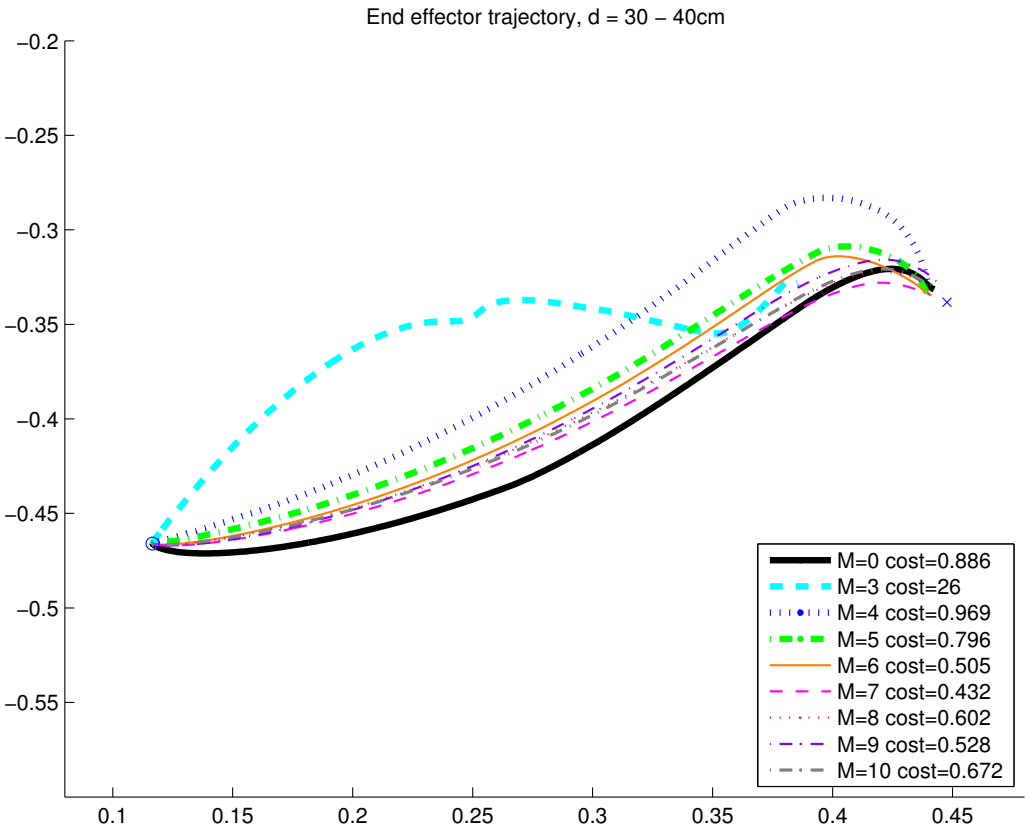
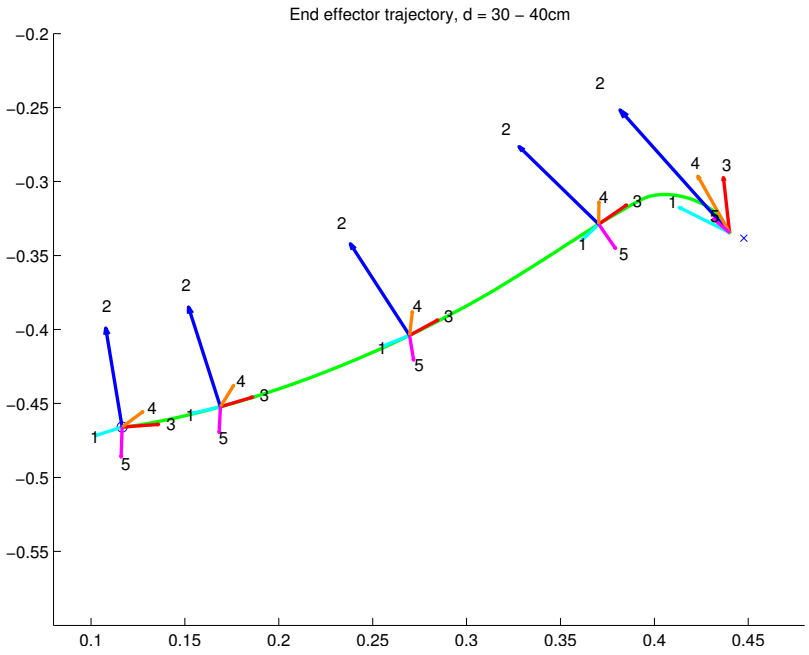
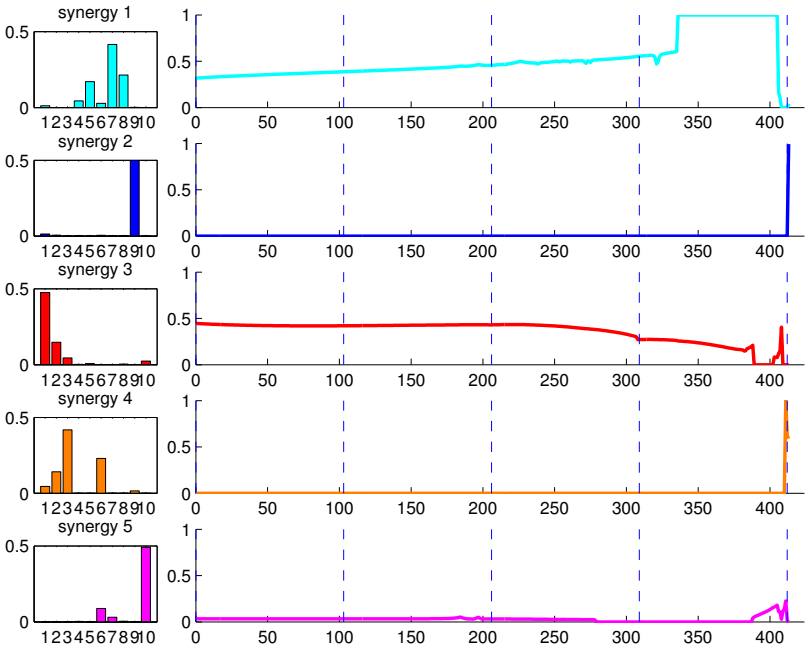


Figure 3-7: Trajectories of a reaching task: A reaching movement of the *original solutions* is shown in dash line (M=0). Movements utilizing different numbers of *goal-related synergies* are shown in corresponding colors (M= K indicates K synergies are utilized). The corresponding cost-to-go values are also shown (smaller values for better trajectory). It is obvious that when more than 3 synergies are utilized, the resulting trajectories are closed to the original solutions (M=0).

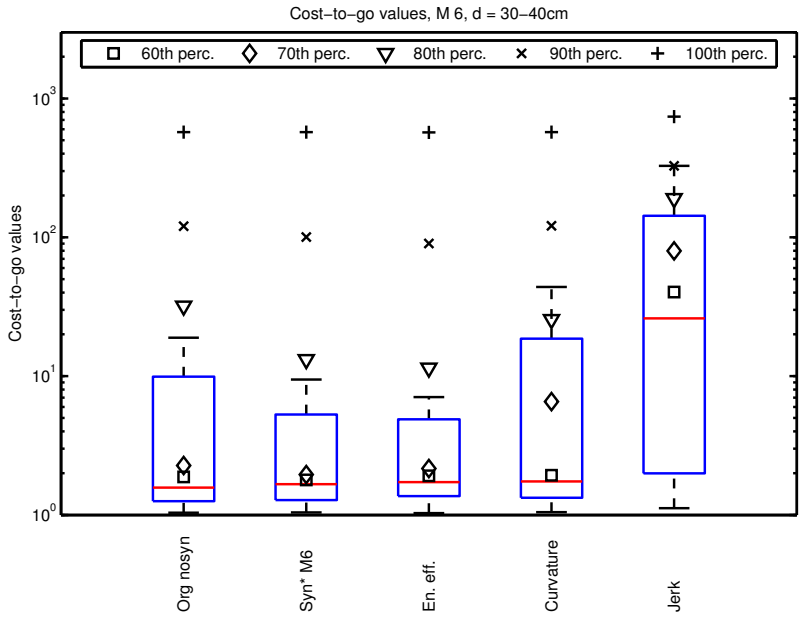


(a) Induced accelerations by synergies

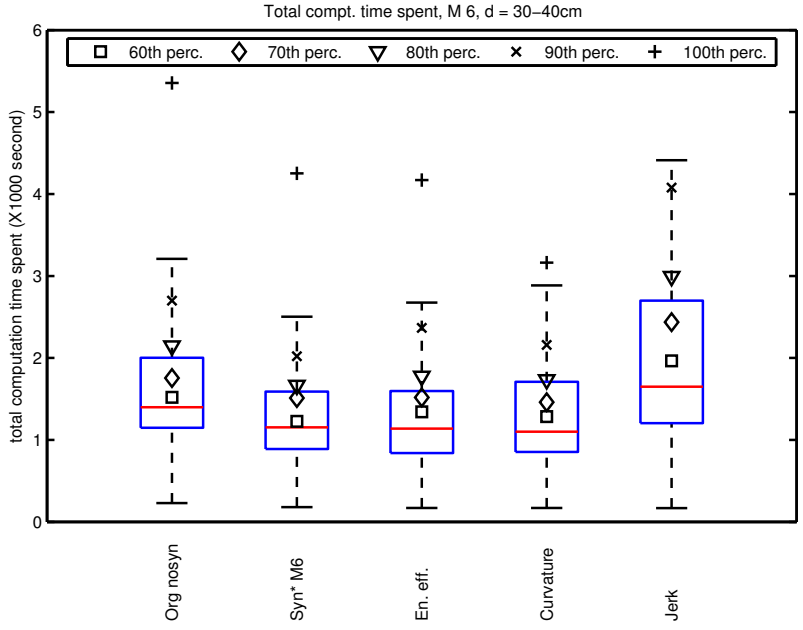


(b) The synergies utilized and the corresponding synergy activation

Figure 3-8: (a) The trajectory shown is the same as the one depicted in Fig. 3-7 when 5 synergies (M = 5) are utilized. The arrows along the trajectory denote the induced accelerations of individual synergies. (b) The goal-related synergies utilized and the corresponding synergy activations of generating the trajectory in (a). It is observed that synergy 3 and synergy 1 dominate the control signal. Synergy 3 and synergy 1 act the opposite and the target, whereas the other synergies act to other directions. Near the end of the movement the activation of synergy 3 decreases, while the other 4 synergy activations increase in order to decelerate the movement.



(a)



(b)

Figure 3-9: Performance of the reaching task (target range: [30-40]cm) utilizing 6 synergies ($M = 6$). Distribution of (a) cost-to-go values and (b) total computation time spent are shown. The columns from the left represent results of *original solutions* (Org nosyn), *synergies-goal solutions* (Syn* M6), *synergies-en solutions* (En. eff.), *synergies-cur solutions* (Curvature) and *synergies-jerk solutions* (Jerk), respectively. It can be observed that utilizing *energy-efficient synergies* performs similarly to utilizing *goal-related synergies*.

optimal control problem. This experiment can also obtain some hints about how to obtain muscle synergies. Because obtaining *goal-related synergies* requires solving for *original solutions* in advance, it is better to discover a way to obtain muscle synergies and then to solve the optimal control problem directly.

The experiment procedure is summarized in Fig. 3-3b. First, three sets of *goal-unrelated synergies* were obtained, namely *energy-efficient synergies* $\mathbf{W}_{en}^{M}$, *curvature synergies* $\mathbf{W}_{cur}^{M}$ and *jerk synergies* $\mathbf{W}_{jerk}^{M}$. Within each of the three sets, $M = 5, ..., 10$ synergies are extracted from 2000 "good" movements generated by GA according to the movement fitness functions V_{en}, V_{cur} and V_{jerk}, respectively, as described in section 3.3.3. Then simulations of the reaching movements utilizing $\mathbf{W}_{en}^{M}$, $\mathbf{W}_{cur}^{M}$ and $\mathbf{W}_{jerk}^{M}$ were carried out, using the same 400 initial and target pairs $\{\boldsymbol{x}_0, \boldsymbol{y}*\}$ of Experiment 1. (Synergies $M = 3, 4$ were not tested because of large reconstruction error in NMF extraction.)

Fig. 3-9 depicts the comparison of the *original solutions*, the *synergies-goal solutions* and the *synergies-en/cur/jerk solutions*. For better illustration, only the results of reaching 100 targets 30-40cm away while utilizing 6 synergies ($M = 6$) are shown. The results of reaching the remaining 300 targets and utilizing different numbers of synergies also have the same trend as depicted in Figs. 3-4b and 3-6b. These results show that utilizing *energy-efficient synergies* can also achieve similar performance compared with utilizing *goal-related synergies*. In contrast, utilizing *curvature synergies* and *jerk synergies* performed badly. Fig. 3-10 depicts a similarity matrix to measures the difference between the *goal-related synergies* $\mathbf{W}_{*}^{M}$ and the *energy-efficient synergies* $\mathbf{W}_{en}^{M}$. It was found that the two sets of synergies *are quite different*. The maximum value of all dot product measure between $\mathbf{W}_{*}^{M}$ and $\mathbf{W}_{en}^{M}$ is about 0.25 for all M. Nevertheless, the *energy-efficient synergies* can still be utilized to achieve the reaching tasks.

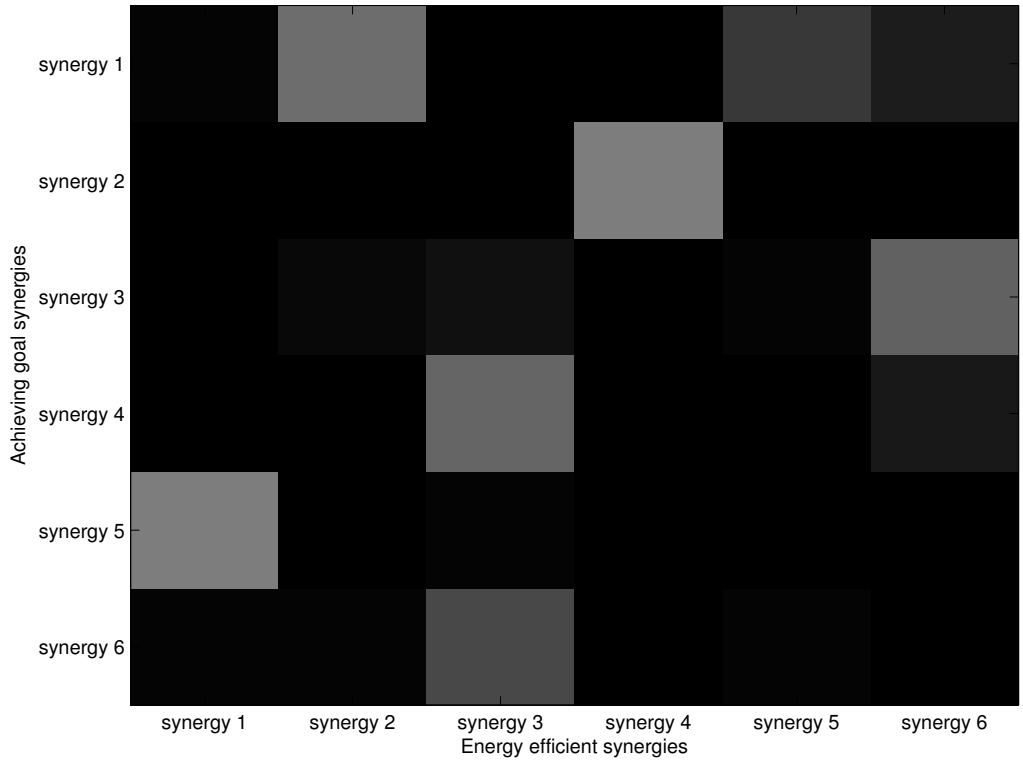


Figure 3-10: Similarity matrix of goal-related synergies and energy-efficient synergies (M = 6). The values of the matrix are the dot product of a goal-related synergy and an energy-efficient synergy. Lighter color has closer value of 1. The synergy 2 of the goal-related synergies and the synergy 4 of the energy are the most similar with dot product value = 0.2489.

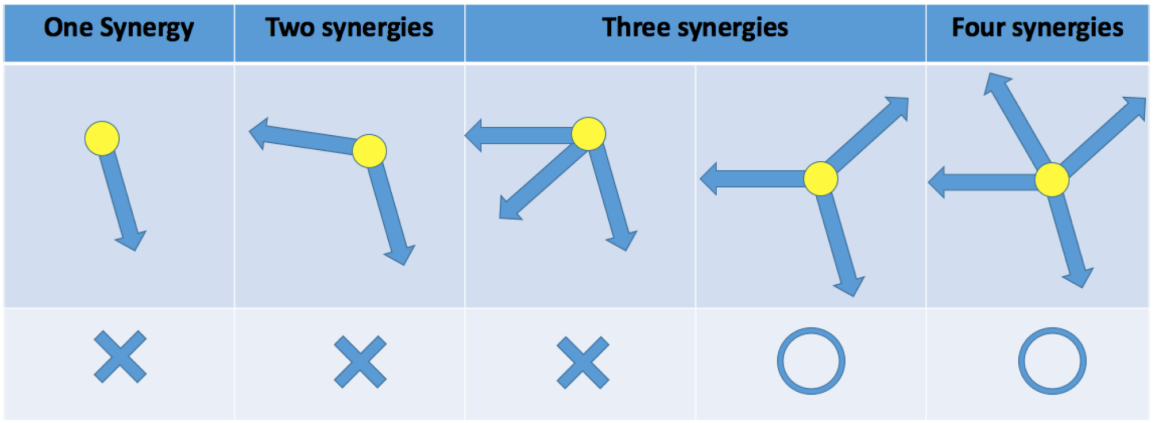


Figure 3-11: Possible moving directions of the robot end-effector when muscle synergies are activated individually. The circles denote the end-effector positions on a 2D task space, and each arrow denotes a moving direction when a muscle synergy is activated. This figure illustrates that at least 3 synergies may be required to move the end-effector in all possible directions on a 2D horizontal task space.

3.5 Discussion

3.5.1 The minimum number of synergies required

At least 3 synergies may be required to move the end-effector in all possible directions on a 2D horizontal task space. A synergy specifies co-activations of all muscles. Given a certain value of synergy activation to a synergy, the muscles correspond to the synergy provide contractile forces on the robot and actuate the end-effector in a certain direction on a 2D task space if the resultant forces on the robot are not canceled out; As illustrated in Fig. 3-11, producing a motion along a line (1D task space) requires at least 2 synergies that move the end-effector in opposite direction, thereby 3 synergies are minimally required to move the end-effector in the 2D task space if combination of any two synergies can move the end-effector in the directions between the two directions that the two synergies can produce.

According to Fig. 3-4a, at least 4 goal-related synergies are necessary. One possible reason is that the moving direction produced by a synergy depends on the robot posture. Therefore, it makes sense that a redundant number of synergies are required to move the end-effector to various target positions.

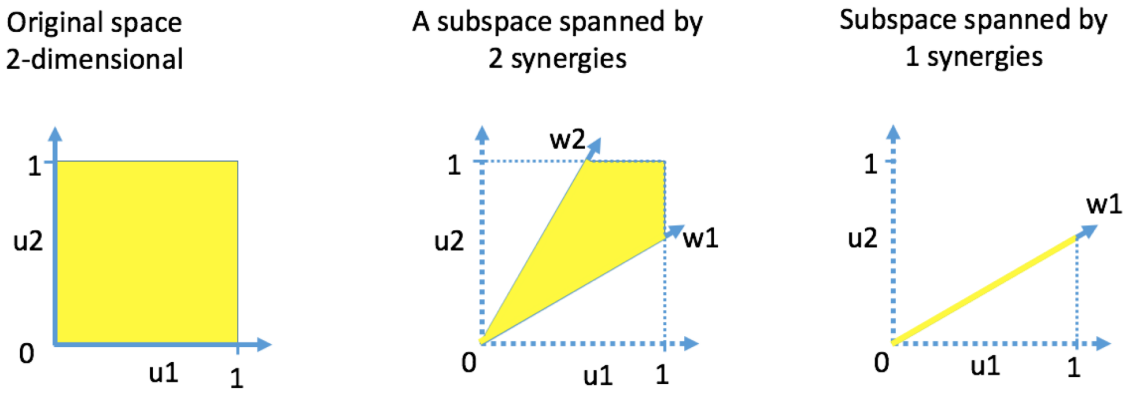


Figure 3-12: Illustrations of the concept of the original control space and subspace spanned by synergies. The admissible set of muscle activation that can actuate the robot is colored in yellow. Muscle activations are bounded such that $0 \leq u1 \leq 1$, $0 \leq u2 \leq 1$, and nonnegative synergy activations (can be larger than 1) are considered in this figure. (Left) A original control space contains a set of admissible muscle activations. (Middle) 2 synergies, w1 and w2, span a subspace within the original space with a smaller area of the admissible muscle activations. (Right) 1 synergy, w1, span a subspace where the admissible muscle activation is along the synergies w1.

3.5.2 Relationship of the cost-to-go values and the number of synergies used

It is expected that the cost-to-go value defined in (3.7) decreases as the number of synergies increased. In this chapter, the robot motion is generated by a well-established "optimal control theory based solver" called iLQG, no matter how the synergies are used or not (Fig. 3-1). It is assumed that the iLQG solver can search for an optimal solution in a given control space such that the cost-to-go value is minimized. The cost-to-go value reflects the control performance defined by the distance from an end-effector target position and velocity at the last time step and the total muscle activation applied to achieve a reaching movement. The smaller the cost-to-go value, the better the control performance. When no synergy is involved (the upper schematic diagram in Fig. 3-1), the control space contains a set of nonnegative and upper bounded muscle activations ($0 \leq \boldsymbol{u} \leq \boldsymbol{u}^{ub}$); This control space is referred as the original control space in the following context. In the original control space, the iLQG controller searches for a set of optimal muscle activations such that the resulting cost-

to-go value is minimized. The number of synergies specifies the control complexity (dimensionality). Fig. 3-12 illustrates the concept of the original control space and the subspace spanned by synergies. When synergies are utilized (the lower schematic diagram in Fig. 3-1), the control space is a subspace of the original control space. This subspace is spanned by a set of muscle synergies. In the subspace, the iLQG controller searches for a set of optimal synergy activations such that the resulting cost-to-go value is minimized. The set of optimal synergy activations is transformed to a set of muscle activations to actuate the robot. Because of the reduction of the dimensionality, the admissible set of muscle activations to actuate the robot thus decreases as fewer synergies are utilized. Therefore, it can be expected that better control performance can be attained when more synergies are utilized; In other words, monotonically decreasing cost-to-go value can be expected when more synergies are utilized. Moreover, under the assumption that the iLQG solver can minimize the cost-to-go function (3.7), it is expected that the cost-to-go value attains the smallest value when no synergy is involved, compared with the case when synergies are utilized. The cost-to-go value obtained in the case of no synergy is referred as the reference value.

The number of synergies does not strongly related to the cost-to-go value. When a set of synergies are utilized, whether the reference value can be attained in a reaching movement does not directly related to the number of synergies used, it is related to whether the optimal muscle activation that corresponds to the reference value, say $\boldsymbol{u}'$, lies within the subspace spanned by the set of synergies. For the case when the set of goal-related synergies is utilized, since the set of synergies is extracted from a set of muscle activations (the *original solution*) that includes $\boldsymbol{u}'$, the synergies span a subspace (of the original control space) that $\boldsymbol{u}'$ lies within; At least using all the synergies as bases, a set of synergy activations can be found by the iLQG controller to approximate $\boldsymbol{u}'$. Moreover, if $\boldsymbol{u}'$ lies within a low-dimensional subspace of the original control space, it is expected that the iLQG controller can found a set of synergy activations to approximate $\boldsymbol{u}'$ using fewer synergies and a cost-to-go value close to the reference value can be attained; In Fig. 3-4a, the cost-to-go values that

closed to the reference values are attained when 4 or more synergies are utilized, implying that the optimal muscle activations of all the reaching movements lie in a low-dimensional control space (manifold). In contrast, for the case when the set of random synergies is used (Fig. 3-5), since the synergies are not orthogonal vectors and the muscle activations are nonnegative and upper bounded, the admissible set of muscle activations to actuate the robot is smaller than, with very small possibility equal to, that of the original control space; The optimal muscle activation $\boldsymbol{u}'$ does not necessarily lie within the subspace spanned by the set of random synergies, thus the cost-to-go value is unlikely to be attained closed to the reference value even all 10 synergies are used. The monotonically decreasing trend of the cost-to-go value in Fig. 3-5 may be caused by such reduction of the admissible set of muscle activations to actuate the robot as fewer synergies are utilized, as illustrated in Fig. 3-12. For the case when the the set of energy-efficient synergies is utilized, it is observed in Fig. 3-4b and Fig. 3-9a that utilizing 5 or more synergies attain cost-to-go values close to the reference values, implying that the set of energy-efficient synergies spans a subspace of the original control space where the optimal muscle activations of all the reaching movements lie within.

3.5.3 Determining the best number of synergies

Because the main purpose of adopting synergies in this thesis is to reduce the control complexity by means of dimensionality reduction, the best number of synergies is regarded as the minimum number of synergies that can achieve a certain level of task performance measured by particular performance indexes. In this chapter, reducing the number of synergies results in the decrease of the computation time spent.

The best number of synergies to achieve the reaching task in this chapter can be determined based on the cost-to-go values that reflect the control performance of the synergies-based iLQG controller. As mentioned in section 3.5.2, the cost-to-go value is expected to be monotonically decreasing (e.g. Fig. 3-4a) with more synergies utilized, with a trade-off of an increase of computation time spent (e.g. Fig. 3-6a) due to the increase of control dimensionality. Moreover, if the set of

muscle activations corresponding to the reference value lies within a low-dimensional subspace that can be spanned by fewer synergies, the cost-to-go value is expected to drop to a level similar to the reference values in the first few synergies (e.g. Fig. 3-4a); The reference cost-to-go value is the cost-to-go value obtained where the iLQG controller searches for solutions in the original control space in full dimensionality. (corresponding to the upper schematic diagram in Fig. 3-1). The best number of synergies can be determined by looking for the minimum number of synergies that has the cost-to-go value close to the reference value. For instance, it can be observed from Fig. 3-4a the best number of goal-related synergies that compromises between the control performance and the computation time spent is 4.

In general, to determine the best number of synergies compromising the control complexity (dimensionality) and the control performance of a synergies-based controller in a case, it is needed to 1) define a performance index to measure the control performance, 2) design a criteria according to the performance index, and 3) design an algorithm to determine the best number of synergies based on the criteria.

The performance index can be, indeed should be, defined to match the objective of the analysis. In this chapter, the objective is to investigate the feasibility of muscle synergies in generating reaching movements by means of solving an optimal control problem. The cost-to-go function defined in (3.7) can quantitatively assess the feasibility of a set of muscle synergies employed in the synergies-based iLQG controller to generate a reaching movement, according to the distance to a target at the last time step and the total muscle activations applied during a reaching movement, such that the smaller the cost-to-go value, the closer to the target. If it is suggested to put more concern on the control assumption, larger weight $\mathbf{R}$ on the term related to the muscle activations should be used in the cost-to-go function in (3.7).

Having the performance index that measures the control performance of a controller, a particular criterion or criteria can be defined for the determination of the best number of synergies based on the (expected) characteristics of the performance index. For example, for the reaching task in this chapter, it is expected that the curve of the cost-to-go values is monotonically decreasing (e.g. Fig. 3-4a) when

more synergies are used. Moreover, the cost-to-go value will drop to a level similar to the reference cost-to-go value is expected to appear, if the set of optimal muscle activations corresponding to the reference cost-to-go value lies in a low-dimensional subspace of the original control space. Based on such characteristics of the curve of the cost-to-go values, the criterion for determining the best number of synergies can be defined as the minimum number of synergies that has the cost-to-go value differing from the reference value smaller than a certain threshold value.

According to the criterion or criteria defined, one can determine the best number of synergies from the results of the performance index when different number of synergies are utilized. A simple approach can be obtaining the results of the performance index for every number of synergies, plotting a curve of the performance index against the number of synergies utilized, and finally determining the best number of synergies from observation according to the criterion or criteria; As it can be observed from Fig. 3-4a that minimum number of synergies is 4. If it can be expected that the value of the performance index will improve as more synergies utilized, one can take an incremental approach that obtains the results of the performance index from the case of the smallest number of synergies, incrementally increases the number of synergies, and terminates until the defined criterion or criteria are satisfied; If the objective of an experiment is to determine the best number of synergies so that the control performance of every number of synergies is not of interest, this incremental approach can save the time to carry out experiments.

3.6 Summary

In this chapter, the feasibility of muscle synergies in controlling musculoskeletal robotic system by optimal control was investigated. The simulations of reaching movements of a human-like robotic arm were analyzed. Two types of muscle synergies having different properties were utilized in simulations, namely *goal-related synergies* and *goal-unrelated synergies*. The former type has the properties of *goal-related* because they are extracted from the optimal solutions of achieving some task goals.

The latter type is *goal-unrelated* because they are extracted from some "good" movements which are optimized according to three specific goal-unrelated criteria, namely *energy-efficient synergies, curvature synergies and jerk synergies*. It was verified that muscle synergies can facilitate the solving of the optimal control problem utilizing *goal-related synergies*. Simulations show that the reaching task can still be achieved, with the same level of success rate and less computation expense. Furthermore, it was investigated what properties muscle synergies should have by analyzing the performance of utilizing *goal-unrelated synergies*. The reaching task could still be achieved when *energy-efficient synergies* are utilized. These results imply that in order to achieve the goal-directed task, synergies do not necessarily need to have *goal-related* properties. Synergies having other properties such as *energy-efficient* can also be utilized to achieve the goal-directed task. The results here also suggest that it is possible to obtain synergies by other methods rather than from solving the computationally expensive optimal control problem.

Chapter 4

Extracting muscle synergies from random movements for low-dimensional task-space control

This chapter studies the extraction of muscle synergies given a data sample of randomly parameterized control signals, without prior knowledge of robot dynamics. In most literature, it is usually assumed that muscle synergies can be extracted from a given data sample with statistical regularities. In reality, this assumption may not be satisfied. A method for extracting muscle synergies with the aid of system identification techniques is proposed. In the proposed method, a forward dynamics model, which predicts the robot's end-effector accelerations from inputs of joint configurations, joint velocities and control signals, is estimated from a data sample of end-effector movements that are generated by randomly parameterized control signals. Using the forward dynamics, a set of optimal control signals that produces the same end-effector accelerations using minimum control effort is estimated by quadratic programming. The required muscle synergies are extracted from the set of control signals estimates. A kernel-based regression technique is adopted to estimate the forward dynamics, as well as the inverse dynamics, which transforms the joint angles, joint velocities and end-effector accelerations (or joint accelerations) to required control signals (i.e. activations of muscles synergies) that achieved the acceleration. A

sliding mode controller is also derived to follow a desired end-effector trajectory, for a class of musculoskeletal systems with the dimensionality of the control input larger than that of the joints and that of the end-effector. The sliding mode controller is a data-driven based controller which consists of the robot inverse dynamics, a feedback control term and a null space control term. The null space control term is responsible for joint stabilization and achieving a secondary task goal, without interfering accuracy of the end-effector. In a simulation evaluation of a human-like robotic arm actuated by 10 muscles, it was shown that muscle synergies could be extracted from the set of control signals estimates, but could not be extracted from the randomly parameterized control signals. In a control task, it was demonstrated that the end-effector could follow a desired figure of "8" trajectory utilizing the extracted muscle synergies, where the control dimensionality was reduced 10 to 5. The null space control also worked well to keep all the joints away from the joint limits, without affecting the tracking accuracy of the end-effector. It was also shown that the control dimensionality could be reduced. It was achieved by first track a set of desired end-effector trajectories spread over the workspace utilizing the current extracted muscle synergies. Then a new set of muscle synergies was extracted from the control signals of the resulting tracking movements. It was found that the figure of "8" trajectory could be tracked when four muscle synergies were utilized. The proposed method provides a solution to the question of how to extract muscle synergies given data sample of control signals without statistical regularities.

4.1 Introduction

Realizing human-like movements and behavior has been one of the most interesting and challenging problems in the fields of robotics. Human-like musculoskeletal robots are usually designed with high dimensionality and redundancy in both control space and joint space, in order to mimic human manipulation ability. However, controlling such complex robots is a challenging problem. One approach is to simplify the control complexity by dimensionality reduction techniques.

In the field of human motor control, muscle synergies hypothesis has been considered as a solution to the degrees-of-freedom (DOF) problem [13, 117, 118, 125]. It has been reported that muscle synergies originate from a human optimal control policy called the minimal intervention principle [121]. This optimal policy focuses control on task-relevant variables and leaves redundant variables uncontrolled as far as possible [68, 126]. In order to reduce the control dimensionality and mimic human task-space control behavior, this chapter focuses on 1) how to extract and 2) utilize muscle synergies to achieve task-space control for human-like musculoskeletal robots.

In many studies of low-dimensional control, muscle synergies were estimated from control signals of given optimal data sources such as human movement data [9, 127], or optimized data with given robot's dynamics model [40, 96]; These data samples possess inherent statistical regularities the signals in one dimension is correlated with the signals in other dimensions. In model-free setting, reinforcement learning was commonly used in which muscle synergies were emerged [128, 129]. However, in common reinforcement learning techniques, a well-designed reward function and initial conditions are usually needed for good learning performance. Therefore, it is also important that a robot can extract muscle synergies from data sample that possesses no statistical regularities without prior knowledge.

This chapter aims at extracting muscle synergies from data sample without statistical regularities, and to utilize the extracted muscle synergies to achieve low-dimension control, for general manipulation of musculoskeletal robots. The control of a human-like robot arm is studied. Under the assumption that large enough samples of random reaching-like movements possess control skills for general manipulation tasks, it is proposed to extract muscle synergies from random reaching-like movements, which are not optimized with respect to minimum control effort. Given enough samples of such reaching-like movements without statistical regularities, a subset of movements is selected according to the end-effector efficiency. The forward dynamics of the robot is then estimated from the subset of data. The corresponding optimal control signals are estimated utilizing the forward dynamics estimates. These estimated optimal control signals are utilized as the source for extracting mus-

cle synergies and the corresponding low-dimensional control signals for the subset of reaching-like movements. The subset of reaching-like movements is regenerated utilizing the low-dimensional control signals. Then the regenerated subset of reaching-like movements is used to estimate the robot's inverse dynamics in the lower dimensional control space for control.

In controlling musculoskeletal robots, estimating robot's dynamics from data is a promising approach to avoid modeling difficulties [130, 131]. Usually, estimation of forward dynamics has no problem because it is a many-to-one mapping. Conversely, due to the redundancy in control space, the inverse dynamics is generally a one-to-many mapping. Therefore, the estimation problem is globally ill-posed and difficult to solve. It is demonstrated that the inverse dynamics could be estimated directly from data, when using the estimated optimal control signals as the training data source. The estimations of the forward dynamics and the inverse dynamics, are formulated into a regression problem within a kernel learning framework. Kernel methods have been widely applied to handle nonlinear and high-dimensional data [132, 133].

In this chapter, a novel method is proposed to extract muscle synergies from optimal control signals estimated from data sample without statistical regularities without prior knowledge of robots for low-dimensional control. In the present work, the estimated inverse dynamics is employed in a sliding controller to achieve task space control. Sliding control is well-known for its robustness, maintaining good control accuracy in the presence of modeling inaccuracies. While sliding control has been studied for standard "square" systems [134] (in which the DOF equals to the number of actuators) and underactuated systems [135], a sliding controller is proposed for applications to overactuated systems such as musculoskeletal system. The control law has a null space control term for stabilization in joint space and achieving secondary goals, without interfering the performance in task space. It is shown that using the estimated inverse dynamics provides maximum capacity for applying the null space control within the admissible control range.

The work in this chapter relates to learning-based control of redundantly actuated robots. In [129], the action space was updated through lower dimensional latent variables in a reinforcement learning process, without the need of applying dimensionality reduction beforehand. They achieved a 3D reaching task using a full-torso musculoskeletal robot. It was needed to start over the learning process when achieving new tasks, which is common in reinforcement learning. Learning-based control from random movements can also be related to "motor babbling" [136], which is an exploratory motor learning process in infants. In [137], an ill-posed robot's inverse kinematics was directly estimated from data in a "trying to reach goals (Goal babbling)" process, in which data samples were weighted according to the redundancy in joint space. Their work aimed at resolving redundancy problem and bootstrap learning. The work here can also be considered as obtaining low-dimensional skills in a motor babbling process. From random movements samples, the work presented here estimates the ill-posed inverse dynamics directly at the control level. In [138] the Jacobian matrix relating the velocities between joint space and task space was iteratively estimated and then its inverse was applied to task space in open-loop motion rate control. Similar studies of task-space control using inverse dynamics estimation can be found in [130, 131]. They solved a globally ill-posed inverse dynamics estimation by estimating the mapping locally, which is well defined in the vicinity of the current robot configuration. In [130], the local control torques were computed in individual local models by averaging the local related data, and then combined to produce the final control torque weighting by local forward dynamics models. In [131], the control torque was given by an inverse dynamics mapping which was updated on-line. The kernel-based estimation formulation proposed here can be considered as a natural generalization of their estimation formulation.

This chapter organizes as follows. First, the problem being studied is defined in section 4.2. The formulation of estimating the robot's forward and inverse dynamics is then introduced in section 4.3. Section 4.4 briefly reviews the concept of muscle synergies, and describes the extraction of the muscle synergies from random reaching-like movement data. In section 4.5, a sliding controller for task-space control is

provided. Section 4.6 explains that using optimal data source enables estimation of inverse dynamics. Finally, section 4.7 provides numerical evaluations of the control performance and analysis of dimension reduction.

4.2 Problem statement

The goals of this chapter are to extract muscle synergies without optimal data given, and to achieve low-dimensional task-space control for general manipulation of a musculoskeletal robot. In particular, the optimality of control signals is restricted to *minimum control effort* measured by the Euclidean norm. Moreover, it is assumed that mathematical model the robot is not given. In order to achieve these goals, the followings are the key issues to be accomplished:

1. Given training samples of random reaching-like movements, extract the muscle synergies that can be utilized to achieve general manipulation.

2. For manipulation, establish a robust task-space tracking control law that controls the end-effector of the robot to follow a desired trajectory in the Cartesian coordinates (task-space). The control law should allow modeling uncertainties, and include a null space control for joint stabilization and achieving secondary goals.

3. Estimation of both the forward and inverse dynamics of the robot, given the kinematics data of the joints and the end-effector, and the control input.

The outline of the extraction of muscle synergies and its utilization for low-dimensional control is illustrated in Fig. 4-1.

Throughout this chapter the following second order continuous time overactuated system is considered

$$\begin{aligned} \ddot{\boldsymbol{x}}(t) &= f(\tilde{\boldsymbol{x}}(t)) + g(\tilde{\boldsymbol{x}}(t))\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= h(\boldsymbol{x}(t)) \\ \ddot{\boldsymbol{y}}(t) &= \alpha(\tilde{\boldsymbol{x}}(t)) + \beta(\tilde{\boldsymbol{x}}(t))\boldsymbol{u}(t) \end{aligned} \tag{4.1}$$

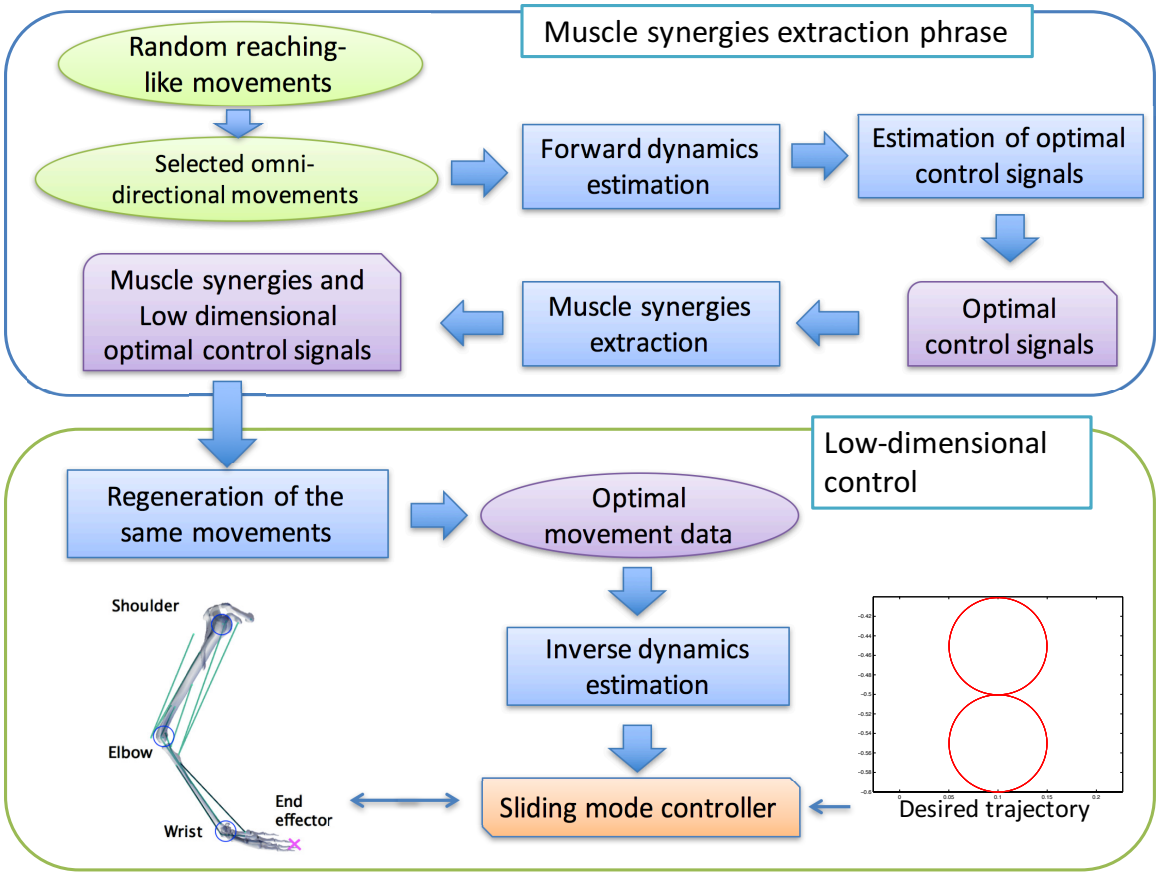


Figure 4-1: Outline of the extraction of muscle synergies and its application for low-dimensional control. (From upper left) The robot's forward dynamics is estimated from a given data sample of omnidirectional movements, which are selected according to the end-effector efficiency. The estimated forward dynamics is then utilized to approximate the corresponding optimal control signals (minimum control effort). From the estimated optimal control signals, muscle synergies and the corresponding low-dimensional control signals are extracted. Then, the same movements are regenerated using the low-dimensional control signals. The regenerated movements are the data source to estimate the robot's inverse dynamics in lower dimensionality. The estimated inverse dynamics and the muscle synergies are utilized in a sliding controller, which control the robot to follow a desired trajectory.

where $\boldsymbol{x} \in \Re^{n^x}$, $\dot{\boldsymbol{x}} \in \Re^{n^x}$ and $\ddot{\boldsymbol{x}} \in \Re^{n^x}$ are the joint angles, velocities and accelerations, respectively. Let $\tilde{\boldsymbol{x}} = [\boldsymbol{x}^T, \dot{\boldsymbol{x}}^T] \in \Re^{2n^x}$ to simplify the notation where T denotes the transpose operation. $\boldsymbol{u} \in \Re^{n^u}$ are the non-negative control signals of actuators with dimension $n^u \geq n^x$ bounded by $\mathbf{0} \leq \boldsymbol{u} \leq \boldsymbol{u}^{ub}$. $\boldsymbol{y} \in \Re^{n^y}$ is the output in task space. f, g , α and β are continuous functions of $\tilde{\boldsymbol{x}}$. h is an observer function. $\alpha(\tilde{\boldsymbol{x}}) \in \Re^{n^y}$ and $\beta(\tilde{\boldsymbol{x}}) \in \Re^{n^y \times n^u}$ are obtained after differentiating $\boldsymbol{y} = h(\boldsymbol{x})$ with respect to time accordingly. f, g, h, α and β are assumed to be unknown. In the following context, the input arguments of all functions f, g, h, α and β, and the time index t are dropped for compact notations. Non-negative control signals were considered because most musculoskeletal systems are actuated by muscles that provide only contraction forces. For simplicity, the dynamics of the actuators were not considered and the joint torque $\boldsymbol{\tau}$ is assumed to be linearly related to the control signals $\boldsymbol{u}$:

$$\boldsymbol{\tau} = \boldsymbol{\Xi}(\tilde{\boldsymbol{x}})\boldsymbol{u} \tag{4.2}$$

where $\boldsymbol{\Xi}(\boldsymbol{x}, \dot{\boldsymbol{x}}) \in \Re^{n^x \times n^u}$ is a matrix of nonlinear functions. Joint torque induced by a simple human muscle model with assumption of inextensible tendon such as the rigid-tendon models in [102] can be described by (4.2). However, common human muscle models having dynamics of contraction [103, 104] introduce unobservable states into the robot dynamics are out of the scope of the present work. The above system was studied in [20]. The planar, 3-DOF, human-like robotic arm actuated by 10 muscles in the Appendix A was used as the simulation platform.

4.2.1 Collecting omnidirectional reaching-like movement data

In chapter 3, it is found that the muscle synergies extracted from optimal goal-directed reaching movements could be utilized for lower dimensional control of similar but not the same goal-directed reaching movements. Therefore, intuitively these muscle synergies can be considered as possessing control skills for general manipulation. In this subsection, the generation of similar reaching-like movements is described. In present study, reaching-like movements were generated by applying the following

Gaussian-like control signal

$$u_m(t) = A_m \exp\left(-\frac{t - c_m}{d_m}\right) \tag{4.3}$$

to the mth actuator at time t, with the end-effector initially resting within the robot's workspace. A group of control signals generated by (4.3) is depicted in Fig. 4-4a. Each movement ended at the same time $t = T$, and started with randomly initialized joint angles $\boldsymbol{x}$ and the parameters A_m, c_m and d_m.

After generation of the reaching-like movements, a subset of the movements is selected for the extraction of muscle synergies. It is usually desirable to manipulate the end-effector with as less control effort as possible. Therefore, it is attempted to select movements in which the end-effector moved most efficiently along the x axis and y axis in the Cartesian plane. Due to the physical structure of the robot, the end-effector may move more efficiently in some particular directions or some particular areas in the workspace. In order to obtain an evenly distributed data set, the workspace is divided into equally sized regions. In each region, an equal number of movements starting from within the same region are selected, according to the end-effector efficiency. The efficiency of a movement in each of the four perpendicular directions was assessed by the ratios of the end-effector displacement to the total control input:

$$\begin{aligned} J_{x+} &= \frac{z_x(t=T) - z_x(t=0)}{\int_{t=0}^{t=T} \|\boldsymbol{u}(t)\|_2}, \quad J_{x-} = -J_{x+} \\ J_{y+} &= \frac{z_y(t=T) - z_y(t=0)}{\int_{t=0}^{t=T} \|\boldsymbol{u}(t)\|_2}, \quad J_{y-} = -J_{y+} \end{aligned} \tag{4.4}$$

where $\boldsymbol{y} = [z_x, z_y]^T$ is the Cartesian coordinates of the end-effector position, t indicates the time and T is the end time of each movement. $\|\cdot\|_2$ denotes Euclidean norm. The ratios indicate better efficiency with larger values. Keeping an even distributed data set is important to achieve good control performance because this subset of movements is subsequently used as the training data for estimating the inverse dynamics. With fixed number of movements to be selected in each region inside the grid, choosing smaller regions ensures a more evenly distributed data set, but also requires more

random movements samples and results in larger training data set. The control signals in the selected subset of movements are denoted as $\boldsymbol{u}_i^{selected}$. Because they are not the minimum norm control signals to achieve end-effector acceleration $\ddot{\boldsymbol{y}}_i$ at state $\tilde{\boldsymbol{x}}_i$. Directly performing PCA on $\boldsymbol{u}_i^{selected}$ does not give a significant performance of dimension reduction in general.

4.3 Estimation of nonlinear affine system

This section describes the estimation of the forward dynamics for muscle synergies extraction, and the inverse dynamics for control. This can be considered as the estimation of unknown nonlinear functions $a(\boldsymbol{\chi})$ and $b_k(\boldsymbol{\chi})$ in the following affine system

$$\nu = a(\boldsymbol{\chi}) + \sum_{m=1}^{r} b_m(\boldsymbol{\chi})\omega_m, \tag{4.5}$$

given N data tuples $\{\nu_i, \boldsymbol{\chi}_i, \boldsymbol{\omega}_i\}_{i=1}^N$, ν_i, $\boldsymbol{\chi}_i \in \Re^{n\chi}$ and $\boldsymbol{\omega}_i = [\omega_{i1}, ..., \omega_{ir}]^T \in \Re^r$ are the scalar output, and the two input vectors, respectively. Denote $\boldsymbol{\omega}^m \in \Re^N$ be the vector having all N data of the mth dimension of $\boldsymbol{\omega}$.

Assume that the nonlinear functions can be approximated by

$$\begin{aligned} a(\boldsymbol{\chi}) &\approx \boldsymbol{w}_a^T \phi(\boldsymbol{\chi}) + \theta_0^a \\ b_m(\boldsymbol{\chi}) &\approx \boldsymbol{w}_{b_m}^T \phi(\boldsymbol{\chi}) + \theta_0^{b_m} \end{aligned} \tag{4.6}$$

with a nonlinear mapping $\phi : \boldsymbol{\chi} \mapsto \phi(\boldsymbol{\chi})$ which maps $\boldsymbol{\chi}$ to an F-dimensional feature (Hilbert) space $\mathcal{F}$ having inner product $k(\boldsymbol{\chi}_p, \boldsymbol{\chi}_q) = \langle \phi(\boldsymbol{\chi}_p), \phi(\boldsymbol{\chi}_q) \rangle$. $\boldsymbol{w}_a \in \Re^F$, $\boldsymbol{w}_{b_m} \in \Re^F$, θ_0^a and $\theta_0^{b_m}$ are the coefficients to be determined. Given N data tuples $\{\nu_i, \boldsymbol{\chi}_i, \boldsymbol{\omega}_i\}_{i=1}^N$, N equations can be formulated

$$\boldsymbol{\nu} \approx \boldsymbol{\Phi}^T \boldsymbol{w}_a + \sum_{m=1}^{r} \mathbf{D}_m \boldsymbol{\Phi}^T \boldsymbol{w}_{b_m} + \tilde{\boldsymbol{\Omega}}^T \boldsymbol{\theta}_0 \tag{4.7}$$

where $\boldsymbol{\nu} = [\nu_1, ..., \nu_N]^T \in \Re^{N\times 1}$, $\boldsymbol{\Phi} = [\phi(\boldsymbol{\chi}_1), ..., \phi(\boldsymbol{\chi}_N)] \in \Re^{F\times N}$, $\mathbf{D}_m \in \Re^{N\times N}$ is a diagonal matrix having vector $\boldsymbol{\omega}^m$, $\tilde{\boldsymbol{\Omega}}^T = [\boldsymbol{\omega}^1, ..., \boldsymbol{\omega}^r, \mathbf{1}_N] \in \Re^{N\times(r+1)}$, $\mathbf{1}_N$ is a

column vector having all entries equal to 1, and $\boldsymbol{\theta}_0 = [\theta_0^{b_1}, ..., \theta_0^{b_r}, \theta_0^a] \in \Re^{r+1}$. Define a slack variable $\boldsymbol{\zeta} = \mathbf{Q}(\boldsymbol{\Phi}^T \boldsymbol{w}_a + \sum_{m=1}^r \mathbf{D}_m \boldsymbol{\Phi}^T \boldsymbol{w}_{b_m} + \tilde{\boldsymbol{\Omega}}^T \boldsymbol{\theta}_0 - \boldsymbol{\nu})$, where $\mathbf{Q}$ is a diagonal matrix with positive entries. The coefficients $\boldsymbol{w}_a$, $\boldsymbol{w}_{b_m}$ and θ_0 are obtained by minimizing the square error and the norm of the coefficients with constraint $\boldsymbol{\Phi}^T \boldsymbol{w}_a + \sum_{m=1}^r \mathbf{D}_m \boldsymbol{\Phi}^T \boldsymbol{w}_{b_m} + \tilde{\boldsymbol{\Omega}}^T \boldsymbol{\theta}_0 - \boldsymbol{\nu} - \mathbf{Q}^{-1}\boldsymbol{\zeta} = \mathbf{0}$:

$$\begin{aligned} &\min \frac{1}{2}\boldsymbol{\zeta}^T\boldsymbol{\zeta} + \frac{\gamma}{2}\left(\boldsymbol{w}_a^T\boldsymbol{w}_a + \sum_{m=1}^r \boldsymbol{w}_{b_m}^T\boldsymbol{w}_{b_m} + \boldsymbol{\theta}_0^T\boldsymbol{\theta}_0\right) \\ &\text{subject to} \\ &\boldsymbol{\Phi}^T \boldsymbol{w}_a + \sum_{m=1}^r \mathbf{D}_m \boldsymbol{\Phi}^T \boldsymbol{w}_{b_m} + \tilde{\boldsymbol{\Omega}}^T \boldsymbol{\theta}_0 - \boldsymbol{\nu} - \mathbf{Q}^{-1}\boldsymbol{\zeta} = \mathbf{0} \end{aligned} \tag{4.8}$$

where γ is a positive scalar. Introducing Lagrangian multipliers $\hat{\boldsymbol{\lambda}}$, (4.8) can be converted to the following unconstrained optimization problem

$$\begin{aligned} &\max_{\hat{\boldsymbol{\lambda}}} \min_{\boldsymbol{\zeta}, \boldsymbol{w}_a, \boldsymbol{w}_{b_m}} E \\ E = &\frac{1}{2}\boldsymbol{\zeta}^T\boldsymbol{\zeta} + \frac{\gamma}{2}\left(\boldsymbol{w}_a^T\boldsymbol{w}_a + \sum_{m=1}^r \boldsymbol{w}_{b_m}^T\boldsymbol{w}_{b_m} + \boldsymbol{\theta}_0^T\boldsymbol{\theta}_0\right) \\ &+ \hat{\boldsymbol{\lambda}}^T\left(\boldsymbol{\Phi}^T \boldsymbol{w}_a + \sum_{m=1}^r \mathbf{D}_m \boldsymbol{\Phi}^T \boldsymbol{w}_{b_m} + \tilde{\boldsymbol{\Omega}}^T \boldsymbol{\theta}_\mathbf{0} - \boldsymbol{\nu} - \mathbf{Q}^{-1}\boldsymbol{\zeta}\right). \end{aligned} \tag{4.9}$$

Solving (4.9) by setting the partial derivatives equal to zero yields

$$\begin{aligned} \hat{\boldsymbol{\lambda}} &= -\gamma\left(\gamma\mathbf{Q}^{-2} + \mathbf{K} + \sum_{m=1}^r \mathbf{D}_m\mathbf{K}\mathbf{D}_m + \tilde{\boldsymbol{\Omega}}^T\tilde{\boldsymbol{\Omega}}\right)^{-1}\boldsymbol{\nu} \\ &= -\gamma\left(\gamma\mathbf{Q}^{-2} + \tilde{\mathbf{K}}^{-1}\right)\boldsymbol{\nu} \\ \boldsymbol{w}_a &= -\frac{1}{\gamma}\boldsymbol{\Phi}\hat{\boldsymbol{\lambda}}, \quad \boldsymbol{w}_{b_m} = -\frac{1}{\gamma}\boldsymbol{\Phi}\mathbf{D}_m\hat{\boldsymbol{\lambda}}, \quad \boldsymbol{\theta}_0 = -\frac{1}{\gamma}\tilde{\boldsymbol{\Omega}}\hat{\boldsymbol{\lambda}}. \end{aligned} \tag{4.10}$$

where the entries of $\mathbf{K}$ at the pth row and qth column are $k(\boldsymbol{\chi}_p, \boldsymbol{\chi}_q)$, $\tilde{\mathbf{K}} = \mathbf{K} + \sum_{m=1}^r \mathbf{D}_m\mathbf{K}\mathbf{D}_m + \tilde{\boldsymbol{\Omega}}^T\tilde{\boldsymbol{\Omega}}$. The estimations of $a(\boldsymbol{\chi}')$ and $b_m(\boldsymbol{\chi}')$ for a test input $\boldsymbol{\chi}'$ are

then given by

$$a(\boldsymbol{\chi}') = -\frac{1}{\gamma}\boldsymbol{k}(\boldsymbol{\chi}')\hat{\boldsymbol{\lambda}} + \theta_0^a$$
$$b_m(\boldsymbol{\chi}') = -\frac{1}{\gamma}\boldsymbol{k}(\boldsymbol{\chi}')\mathbf{D}_m\hat{\boldsymbol{\lambda}} + \theta_0^{b_m}$$

where the entries of $\boldsymbol{k}(\boldsymbol{\chi}') \in \Re^N$ are $k(\boldsymbol{\chi}', \boldsymbol{\chi}_i), i = 1, ..., N$.

The computation of $\hat{\boldsymbol{\lambda}}$ needs the inversion of the matrix $\tilde{\mathbf{K}}$, which has computation cost $\mathcal{O}(N^3)$. The computation is impractical for large N. Sparsification technique is a possible way for practical implementation. Let $\tilde{\boldsymbol{\chi}}' = [\boldsymbol{\chi}'^T, \boldsymbol{\omega}'^T]^T$, it can be shown that

$$\nu(\tilde{\boldsymbol{\chi}}') = \tilde{\boldsymbol{k}}(\tilde{\boldsymbol{\chi}}')\left(-\frac{1}{\gamma}\hat{\boldsymbol{\lambda}}\right) \tag{4.11}$$

where the entries of the vector $\tilde{\boldsymbol{k}}(\tilde{\boldsymbol{\chi}}')$ are computed using the composite kernel

$$\tilde{k}(\tilde{\boldsymbol{\chi}}', \tilde{\boldsymbol{\chi}}_{\boldsymbol{i}}) = k(\boldsymbol{\chi}', \boldsymbol{\chi}_{\boldsymbol{i}}) + \sum_{m=1}^{r} \omega'_m k(\chi', \chi_i)\omega_{im} + [\boldsymbol{\omega}'^T, 1][\boldsymbol{\omega}_i^T, 1]^T. \tag{4.12}$$

with training data $\tilde{\boldsymbol{\chi}}_i, i = 1, ..., N$. Note that (4.11) is the standard form of common nonlinear kernel-based regression. Therefore, the sparsification can be achieved by solving for the spanning coefficients $\underline{\boldsymbol{\lambda}} = -\frac{1}{\gamma}\hat{\boldsymbol{\lambda}}$ using common sparse kernel regression techniques. The software SparseBayes 2.0 [139] is used, which implements the Relevance Vector Machines (RVM) [140, 141], to solve for $\underline{\boldsymbol{\lambda}}$. One of the advantages of RVM is that the regularization constant γ is automatically determined, users only need to choose the proper valid kernel $k(\boldsymbol{\chi}_p, \boldsymbol{\chi}_q) = \langle\phi(\boldsymbol{\chi}_p), \phi(\boldsymbol{\chi}_p)\rangle$.

4.4 Extraction of muscle synergies

In most studies such as [9, 40, 96, 127–129], muscle synergies were extracted from controlled or optimal specific movement data. Intuitively, those synergies can be utilized to control movements similar to the original ones. One of our goals is the extraction of muscle synergies without optimal data given. Moreover, the extracted synergies can be utilized for general manipulation tasks. This section starts by giving

the definition of muscle synergies. Then the generation of reaching-like movements is briefly described. It will be demonstrated that muscle synergies can be extracted from optimal data, which are estimated using the forward dynamics estimated.

4.4.1 Definition

There are many interpretations of muscle synergies in various studies such as time varying synergies [17, 20, 142], or uncontrolled manifold [118, 143]. In this chapter, time-invariant synergies is adopted that a muscle synergy $\boldsymbol{w}_j \in \Re^{n^u}$ represents a synchronous activation of a group of n^u actuators. Applications of time-invariant synergies to robot control can also be found in [144], where robots were manipulated by a feedback controller. Applications of time-varying synergies can be found in [96, 145]. In contrast to time-invariant synergies that possess only spatial information, time-varying synergies are sequences of (control) action which possesses both temporal and spatial information. The presence of temporal information increases the difficulties to implement time-varying synergies to a feedback controller. It is still an open question that which interpretation is the best. Time-invariant synergies are chosen to achieve general manipulation because of it is simpler to implement conventional feedback control techniques. However, controllers (usually open-loop) utilizing time-varying synergies usually have much fewer control parameters. Therefore, they are especially beneficial for controlling simple tasks such as reaching movements. Development of a controller utilizing both time-invariant and time-varying synergies is a promising direction and will be studied in the future. Readers can refer to [119] for more details about different interpretations of muscle synergies.

A control signal $\boldsymbol{u} \in \Re^{n^u}$ is constructed by linear combination of $M \leq n^u$ muscle synergies. Given samples of the control signals, muscle synergies can be extracted by various matrix factorization techniques [125]. In this chapter, Principal Component Analysis (PCA) [108] is used to extract muscle synergies, because PCA is one of the most common dimension reduction techniques and easy to understand. Also, because dimension reduction performance depends on data characteristics, attention is put on development to achieve low-dimensional control in a general manner. Investigation

of different dimension reduction techniques for better performance will be carried out in future work. Through linear matrix factorization, a control signals vector $\boldsymbol{u}$ is approximated by

$$\boldsymbol{u} \approx \sum_{j=1}^{M} \boldsymbol{w}_j a_j + \bar{\boldsymbol{u}} = \mathbf{W}\boldsymbol{a} + \bar{\boldsymbol{u}} \tag{4.13}$$

where $\mathbf{W}$ contains the first $M < n^u$ most important principal components in the columns. $\bar{\boldsymbol{u}}$ is the mean of $\boldsymbol{u}$. a_j is the activation of the jth muscle synergy $\boldsymbol{w}_j$. The muscle synergies extracted by PCA may have negative components. A muscle with negative activation can be considered a muscle which produces extensional force. The linear combination of muscle synergies (5.2) must be non-negative because a muscle can only produce contraction force and bounded above by maximum activation $\boldsymbol{u}^{ub}$. In implementation, the values in $\boldsymbol{u}$ were set to 0 or $\boldsymbol{u}^{ub}$, respectively, when they become negative or larger than $\boldsymbol{u}^{ub}$, .

4.4.2 Extraction of muscle synergies from estimated optimal control signals

Using the forward dynamics in task space (the third equation in 4.1), the corresponding optimal control signals $\boldsymbol{u}_i^{min}$ can be estimated. If the control is not bounded, $\boldsymbol{u}_i^{min}$ is simply given by $\boldsymbol{u}_i^{min} = \beta^{\dagger}(\tilde{\boldsymbol{x}}_i)(\ddot{\boldsymbol{y}}_i - \alpha(\tilde{\boldsymbol{x}}_i))$ where $\mathbf{M}^{\dagger}$ denotes the Moore-Penrose pseudo-inverse of a matrix $\mathbf{M}$. For bounded control, the optimal control signals $\boldsymbol{u}_i^{min}$ are estimated by solving the following quadratic programming problem:

$$\begin{aligned} &\min \quad \frac{1}{2}(\boldsymbol{u}_i^{min})^T \boldsymbol{u}_i^{min} \\ &\text{subject to} \begin{cases} \ddot{\boldsymbol{y}}_i = \alpha(\tilde{\boldsymbol{x}}_i) + \beta(\tilde{\boldsymbol{x}}_i)\boldsymbol{u}_i^{min}, \\ \mathbf{0} \le \boldsymbol{u}_i^{min} \le \boldsymbol{u}^{ub}. \end{cases} \end{aligned} \tag{4.14}$$

The estimation of the functions α and β in the forward dynamics has been described in section 4.3. The key idea is to approximate the optimal control signals $\boldsymbol{u}^{min}$ from the selected omnidirectional movements, with the aid of forward dynamics estimates.

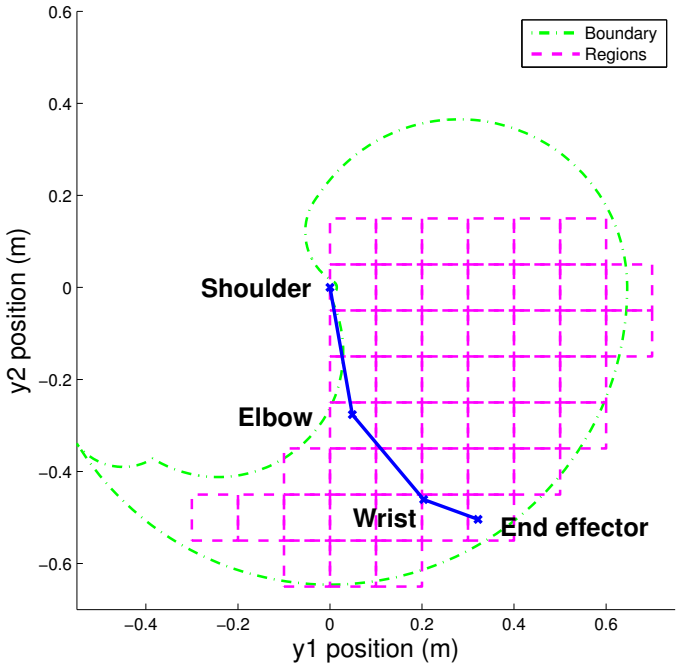


Figure 4-2: The workspace of the robotic arm. Within the square regions, data sample of movements generated by randomly parameterized control signals were collected for synergies extraction. The collected data sample was the source for the result shown in Fig. 4-3.

In order to demonstrate that using the optimal control signals as data source enables dimension reduction, muscle synergies were extracted from reaching-like movements data from the workspace of the robot depicted in Fig. 4-2. 383 of 10000 random reaching-like movements generated by the randomly parameterized control signal defined in (4.3) were selected, from 48 square regions (0.1m X 0.1m each), according to the end-effector efficiency. The set of the omnidirectional movement's control signals is denoted as $\boldsymbol{u}^{selected}$. The corresponding optimal control signals $\boldsymbol{u}^{min}$ were estimated after estimating the forward dynamics using the selected movement data. To compare the performance of the selection process, 383 of the same 10000 reaching-like movements were randomly selected. The control signals $\boldsymbol{u}^0$ were collected and the corresponding optimal control signals $(\boldsymbol{u}^0)^{min}$ were estimated after estimating the forward dynamics using the randomly selected movement data. Fig. 4-3 depicts the variance characteristics of performing PCA on $\boldsymbol{u}^0$, $\boldsymbol{u}^{selected}$, $(\boldsymbol{u}^0)^{min}$ and $\boldsymbol{u}^{min}$. Dimension reduction was observed in the case of estimated optimal control signals

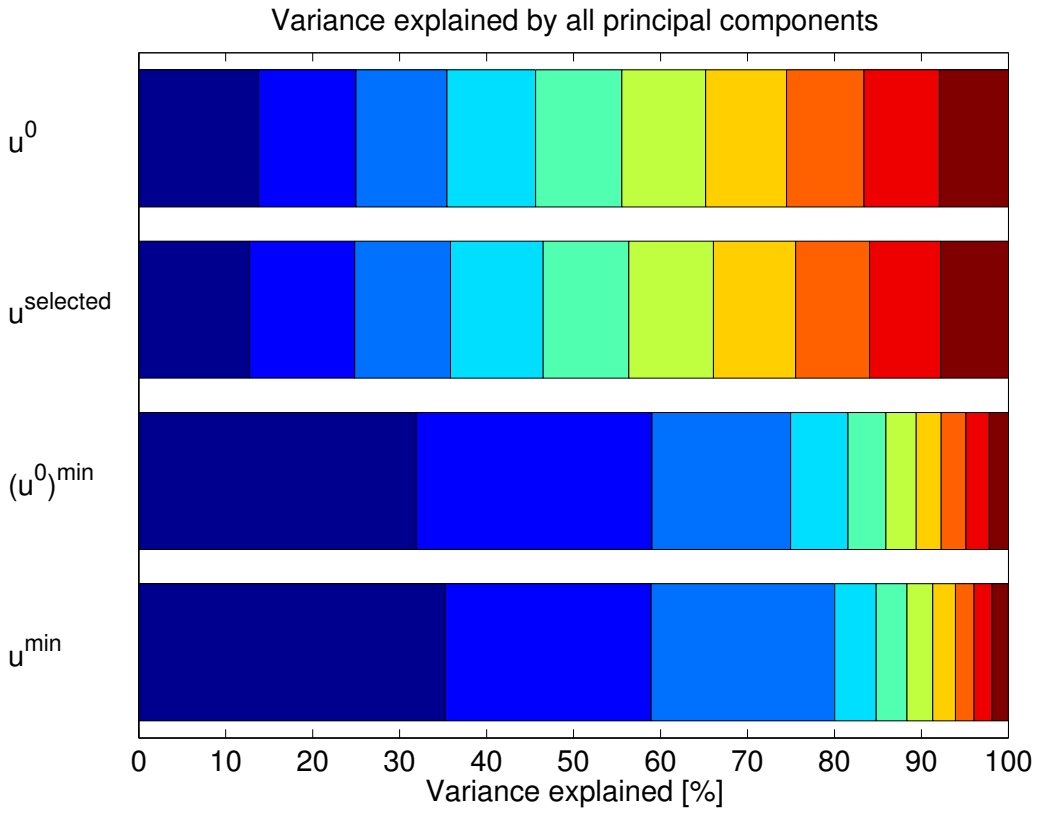


Figure 4-3: Percentage of variance explained by all the principal components of PCA, which was performed on the control signals of (1) $\boldsymbol{u}^0$: control signal from randomly selected reaching-like movements; (2) $\boldsymbol{u}^{selected}$: control signals selected according to the end-effector efficiency; (3) $(\boldsymbol{u}^0)^{min}$: the estimated optimal control signals corresponding to $\boldsymbol{u}^0$ and (4) $\boldsymbol{u}^{min}$: the estimated optimal control signals corresponding to $\boldsymbol{u}^{selected}$.

$(\boldsymbol{u}^0)^{min}$ and $\boldsymbol{u}^{min}$. A better degree of dimension reduction was observed after selection. Because every principal component shares almost the same importance, muscle synergies cannot be extracted from neither the data $\boldsymbol{u}^0$ that without statistical regularities, nor the selected data $\boldsymbol{u}^{selected}$. Recall that because muscle synergies were obtained from random reaching-like movements for general manipulation, the degree of dimension reduction provided by these synergies was expected to be not as good as those synergies obtained from well-optimized/controlled signals of simple movements. It was observed that the degree of dimension reduction is not very significant that 5 PCs explain 90% variance of a 10-dimensional data. In section 4.7 it will show that better degree of dimension reduction can be achieved when controlled movements are the data source.

4.5 A sliding controller for overactuated system

This section describes a controller which computes the control signal to manipulate the end-effector to follow a desired trajectory, using estimates of the inverse dynamics of the overactuated system (4.1). Sliding control technique is adopted for allowing estimation inaccuracies. The controller also has a null space control term for joint stabilization and achieving secondary goals.

Precisely, the problem is find control signals $\boldsymbol{u}(t)$ to control the overactuated system (4.1) to follow a desired trajectory $\tilde{\boldsymbol{y}}(t) = [(\boldsymbol{y}^*(t))^T, (\dot{\boldsymbol{y}}^*(t))^T]^T \in \Re^{2n^y}$ in task space such that

$$\forall t \geq 0, |\boldsymbol{e}| \leq \boldsymbol{\epsilon}_1, |\dot{\boldsymbol{e}}| \leq \boldsymbol{\epsilon}_2, \boldsymbol{\epsilon}_1 > 0, \boldsymbol{\epsilon}_2 > 0 \tag{4.15}$$

with initial condition $\tilde{\boldsymbol{y}}^*(0) = [(\boldsymbol{y}(0))^T, (\dot{\boldsymbol{y}}(0))^T]$, where $\boldsymbol{e} = \boldsymbol{y} - \boldsymbol{y}^*$ and $\dot{\boldsymbol{e}} = \dot{\boldsymbol{y}} - \dot{\boldsymbol{y}}^*$ are the tracking error vectors and $|\cdot|$ is an entry-wise absolute value operator.

Following standard procedure in sliding control [134], let

$$\boldsymbol{s} = \dot{\boldsymbol{e}} + \lambda \boldsymbol{e}, \quad \lambda > 0, \lambda \in \Re. \tag{4.16}$$

The tracking problem (4.15) is achieved by finding a control law which satisfies the sliding condition

$$\frac{1}{2}\frac{d}{dt}(\boldsymbol{s}^T\boldsymbol{s}) \leq -\boldsymbol{\eta}^T|\boldsymbol{s}|, \quad \boldsymbol{\eta} > \mathbf{0}. \tag{4.17}$$

Let $\tilde{\alpha}$ and $\tilde{\beta}$ be the estimates of α and β in (4.1), respectively. Furthermore, assume that the estimation errors are bounded such that $|\alpha - \tilde{\alpha}| \leq \mathbf{A}$ and $|\beta - \tilde{\beta}| \leq \mathbf{B}$, where the entries of $\mathbf{A}$ and $\mathbf{B}$ are non-negative finite real number. Consider the following control law

$$\boldsymbol{u} = \tilde{\beta}^{\dagger}(\ddot{\boldsymbol{y}}^* - \tilde{\alpha}) - \tilde{\beta}^{\dagger}((\boldsymbol{k}_s \circ \mathrm{sgn}(\boldsymbol{s})) + \lambda\dot{\boldsymbol{e}}) + (\mathbf{I} - \tilde{\beta}^{\dagger}\tilde{\beta})\boldsymbol{u}_0 \tag{4.18}$$

where $\tilde{\beta}^{\dagger}$ is the Moore-Penrose pseudo-inverse of $\tilde{\beta}$ and $\mathbf{I}$ is the identity matrix, $\mathrm{sgn}(\cdot)$ is an entry-wise operator which returns 1, -1 and 0 if the corresponding entry is positive, negative or equal to zero respectively, $\circ$ denotes the Hadamard product and $\boldsymbol{u}_0$ is a null space control vector which will be explained in more detail later. In order to satisfy the sliding condition (4.17), one can choose large enough control gain $\boldsymbol{k}_s$ such that

$$\beta\tilde{\beta}^{\dagger}\boldsymbol{k}_s \geq \mathbf{A} + \mathbf{B}|\hat{\boldsymbol{u}}| + \boldsymbol{\eta} \tag{4.19}$$

where $\hat{\boldsymbol{u}} = \tilde{\beta}^{\dagger}(\ddot{\boldsymbol{y}}^* - \tilde{\alpha} - \lambda\dot{\boldsymbol{e}}) + (\mathbf{I} - \tilde{\beta}^{\dagger}\tilde{\beta})\boldsymbol{u}_0$. In other words, the end-effector can follow the desired trajectory $\tilde{\boldsymbol{y}}(t)$, by choosing large enough control gain $\boldsymbol{k}_s$ in the control law (5.13) to compensate the estimation inaccuracies $\mathbf{A}$ and $\mathbf{B}$.

For the proof, consider a Lyapunov-like function

$$V(\boldsymbol{s}, t) = \frac{1}{2}\boldsymbol{s}^T\boldsymbol{s} \geq 0. \tag{4.20}$$

Differentiating with respect to time yields

$$\begin{aligned}\dot{V} =& \boldsymbol{s}^T \dot{\boldsymbol{s}} \\ =& \boldsymbol{s}^T (\ddot{\boldsymbol{y}} - \ddot{\boldsymbol{y}}^* + \lambda \dot{\boldsymbol{e}}) \\ =& \boldsymbol{s}^T (\alpha + \beta \boldsymbol{u} - \ddot{\boldsymbol{y}}^* + \lambda \dot{\boldsymbol{e}}) \\ =& \boldsymbol{s}^T (\alpha + \beta \hat{\boldsymbol{u}} - \tilde{\beta} \hat{\boldsymbol{u}} + \tilde{\beta} \hat{\boldsymbol{u}} - \ddot{\boldsymbol{y}}^* + \lambda \dot{\boldsymbol{e}} - \beta \beta^\dagger (\boldsymbol{k}_s \circ \operatorname{sgn}(\boldsymbol{s}))) \\ =& \boldsymbol{s}^T [(\alpha - \tilde{\alpha}) + (\beta - \tilde{\beta}) \hat{\boldsymbol{u}} - \beta \tilde{\beta}^\dagger (\boldsymbol{k}_s \circ \operatorname{sgn}(\boldsymbol{s}))].\end{aligned} \tag{4.21}$$

Using (4.19) yields

$$\begin{aligned}\dot{V} &\leq \boldsymbol{s}^T [\mathbf{A} + \mathbf{B}|\hat{\boldsymbol{u}}| - (|\mathbf{A} + \mathbf{B}|\hat{\boldsymbol{u}}| + \boldsymbol{\eta}) \circ \operatorname{sgn}(\boldsymbol{s})] \\ &= \boldsymbol{s}^T (\mathbf{A} + \mathbf{B}|\hat{\boldsymbol{u}}|) - |\boldsymbol{s}|^T (\mathbf{A} + \mathbf{B}|\hat{\boldsymbol{u}}|) - \boldsymbol{\eta}^T |\boldsymbol{s}| \\ &\leq -\boldsymbol{\eta}^T |\boldsymbol{s}| \leq 0.\end{aligned} \tag{4.22}$$

From Barbalat's Lyapunov-like lemma, it can be concluded that $\boldsymbol{s} \to \mathbf{0}$ and therefore $\boldsymbol{e} \to \mathbf{0}$.

In order to reduce the effect of chattering, the $\operatorname{sgn}(\cdot)$ function is replaced by a saturation function

$$\boldsymbol{u} = \tilde{\beta}^\dagger (\ddot{\boldsymbol{y}}^* - \tilde{\alpha}) - \tilde{\beta}^\dagger ((\boldsymbol{k}_s \circ \ \operatorname{sat}(\boldsymbol{s}, \boldsymbol{\Psi})) + \lambda \dot{\boldsymbol{e}}) + (\mathbf{I} - \tilde{\beta}^\dagger \tilde{\beta}) \boldsymbol{u}_0 \tag{4.23}$$

where sat is an entry-wise function

$$\begin{aligned}&\operatorname{sat}(s_n, \Psi_n) = s_n \qquad \text{if } \left| \frac{s_n}{\Psi_n} \right| \leq 1 \\ &\operatorname{sat}(s_n, \Psi_n) = \operatorname{sgn}(s_n) \qquad \text{otherwise.}\end{aligned} \tag{4.24}$$

where $n = 1, ..., n^y$ is the dimension index, $\boldsymbol{\Psi} \in \Re^{n^y}$ is vector of positive entries Ψ_n.

In (4.23), the first term corresponds to the feedforward control. The second term corresponds to the feedback control. In the last term $(\mathbf{I} - \tilde{\beta}^\dagger \tilde{\beta})$ projects the null space control $\boldsymbol{u}_0$ to the null space of $\tilde{\beta}$, which only acts in the null space and thus does not affect the performance in the task space. The null space control $\boldsymbol{u}_0$ is utilized

for stabilization in the joint space and to achieve secondary goals, during tracking in the task space. In simulation, the following null space control was employed in order to achieve a secondary goal of avoiding collision at the joint limits x_l^{lb} and x_l^{ub}, $l = 1, ..., n^x$:

$$\boldsymbol{u}_0 = \tilde{g}^{\dagger}[-\mathbf{K}_{v0}\dot{\boldsymbol{x}} - \mathbf{K}_{p0}\boldsymbol{\varepsilon}(\boldsymbol{x})] \tag{4.25}$$

where $\tilde{g}^{\dagger}$ was the Moore-Penrose pseudo-inverse of the estimate of g, $\mathbf{K}_{v0}$ and $\mathbf{K}_{p0}$ were the diagonal matrices with positive entries in their diagonals. $\boldsymbol{\varepsilon}$ was an entry-wise function that the lth entry given by

$$\varepsilon_l = \begin{cases} x_l - (x_l^{lb} + \delta), & x_l^{lb} \leq x_l \leq x_l^{lb} + \delta \\ x_l - (x_l^{ub} - \delta), & x_l^{ub} - \delta \leq x_l \leq x_l^{ub} \\ 0 & \text{otherwise} \end{cases} \tag{4.26}$$

and δ was a positive scalar. One can replace $\boldsymbol{\varepsilon}(\boldsymbol{x})$ to achieve other secondary goals. For example, choosing $\boldsymbol{u}_0 = \tilde{g}^{\dagger}(-\mathbf{K}_{v0}\dot{\boldsymbol{x}} - \mathbf{K}_{p0}(\boldsymbol{x} - \boldsymbol{x}_{rest}))$ has the effect of bringing $\boldsymbol{x}$ to "resting posture" $\boldsymbol{x}_{rest}$.

It is desirable to have small control gains. Briefly speaking, using minimum norm control signals allows larger null space control $\boldsymbol{u}_0$ for achieving secondary goals. Within the admissible control range $\mathbf{0} \leq \boldsymbol{u} \leq \boldsymbol{u}^{ub}$ where $\boldsymbol{u}$ is defined by (4.23), using the estimated optimal control in the feedforward control

$$\beta^{\dagger}(\tilde{\boldsymbol{x}})\,(\ddot{\boldsymbol{y}} - \alpha(\tilde{\boldsymbol{x}})) = \boldsymbol{u}^{min} \tag{4.27}$$

having the minimum absolute values, has the advantage of providing maximum capacity to apply the null space control $\boldsymbol{u}_0$ and the feedback control, since the required control gain $\boldsymbol{k}_s$ in the feedback control becomes the smallest. It is because acting larger null space control $\boldsymbol{u}_0$ ($\boldsymbol{u}_0 \neq \mathbf{0}$) results in larger $|\hat{\boldsymbol{u}}|$ in (4.19), thus larger $\boldsymbol{k}_s$ is needed in the feedback control to satisfy the condition (4.19) in order to achieve tracking under modeling uncertainties. Readers can refer to [116] for more details about null space control in task space control.

4.6 Inverse dynamics estimation

4.6.1 Estimation of the inverse dynamics from optimal control signals

In order to apply the sliding control law (4.23), it is needed to estimate the terms $\beta^{\dagger}\alpha$ and $\beta^{\dagger}$ in the inverse dynamics. One of the common difficulties of inverse dynamics estimation is that the mapping is one-to-many in general. Consider the over-actuated system (4.1) with $\boldsymbol{u}$ unbounded, the inverse dynamics is given by

$$\begin{aligned} \boldsymbol{u} &= \beta^{\dagger}(\tilde{\boldsymbol{x}})(\ddot{\boldsymbol{y}} - \alpha(\tilde{\boldsymbol{x}})) + (\mathbf{I} - \beta^{\dagger}(\tilde{\boldsymbol{x}})\beta(\tilde{\boldsymbol{x}}))\boldsymbol{\xi} \\ &= \boldsymbol{u}^{min} + (\mathbf{I} - \beta^{\dagger}(\tilde{\boldsymbol{x}})\beta(\tilde{\boldsymbol{x}}))\boldsymbol{\xi} \end{aligned} \tag{4.28}$$

where $\boldsymbol{\xi}$ is an arbitrary vector and $\boldsymbol{u}^{min}$ is given by (4.27). This implies that there are many ways to achieve $\ddot{\boldsymbol{y}}$ at the same $\tilde{\boldsymbol{x}}$. The estimation of $\beta^{\dagger}\alpha$ and $\beta^{\dagger}$ is impossible because $\boldsymbol{\xi}$ is unknown. Inspired by the fact that the optimal control $\boldsymbol{u}^{min}$ has the minimum Euclidean norm among all possible solutions[1], the estimation of $\beta^{\dagger}\alpha$ and $\beta^{\dagger}$ is feasible if the training samples are optimal control signals among all possible solutions to achieve $\ddot{\boldsymbol{y}}_i$ at $\tilde{\boldsymbol{x}}_i$ in the sense of minimum Euclidean norm. In this research, optimal control signals are approximated by solving the optimization problem (4.27). Consequently, it can be considered as the case of $\boldsymbol{\xi} \approx 0$ and thus the inverse dynamics mapping can be directly estimated from the estimated optimal control signals $\boldsymbol{u}_i^{min}$.

4.6.2 Regeneration of the same movements using lower-dimensional control signals

Before estimating the task space inverse dynamics mapping $(\tilde{\boldsymbol{x}}, \ddot{\boldsymbol{y}}) \mapsto \boldsymbol{a}$ for low-dimensional control, it is necessary to ensure that each $\ddot{\boldsymbol{y}}_i$ is the outcome of applying the low-dimensional control signal $\boldsymbol{a}_i$ at $\tilde{\boldsymbol{x}}_i$ in the training data set $\{\boldsymbol{a}_i, \tilde{\boldsymbol{x}}_i, \ddot{\boldsymbol{y}}_i\}, i = 1, ..., N$. To this end, the same omnidirectional movements are regenerated, by ap-

[1]Since the minimum Euclidean norm solution of linear system $\mathbf{M}\boldsymbol{z} = \boldsymbol{b}$ with non-unique solutions is given by $\boldsymbol{z} = \mathbf{M}^{\dagger}\boldsymbol{b}$ where $\mathbf{M}^{\dagger}$ is the Moore-Penrose inverse of $\mathbf{M}$.

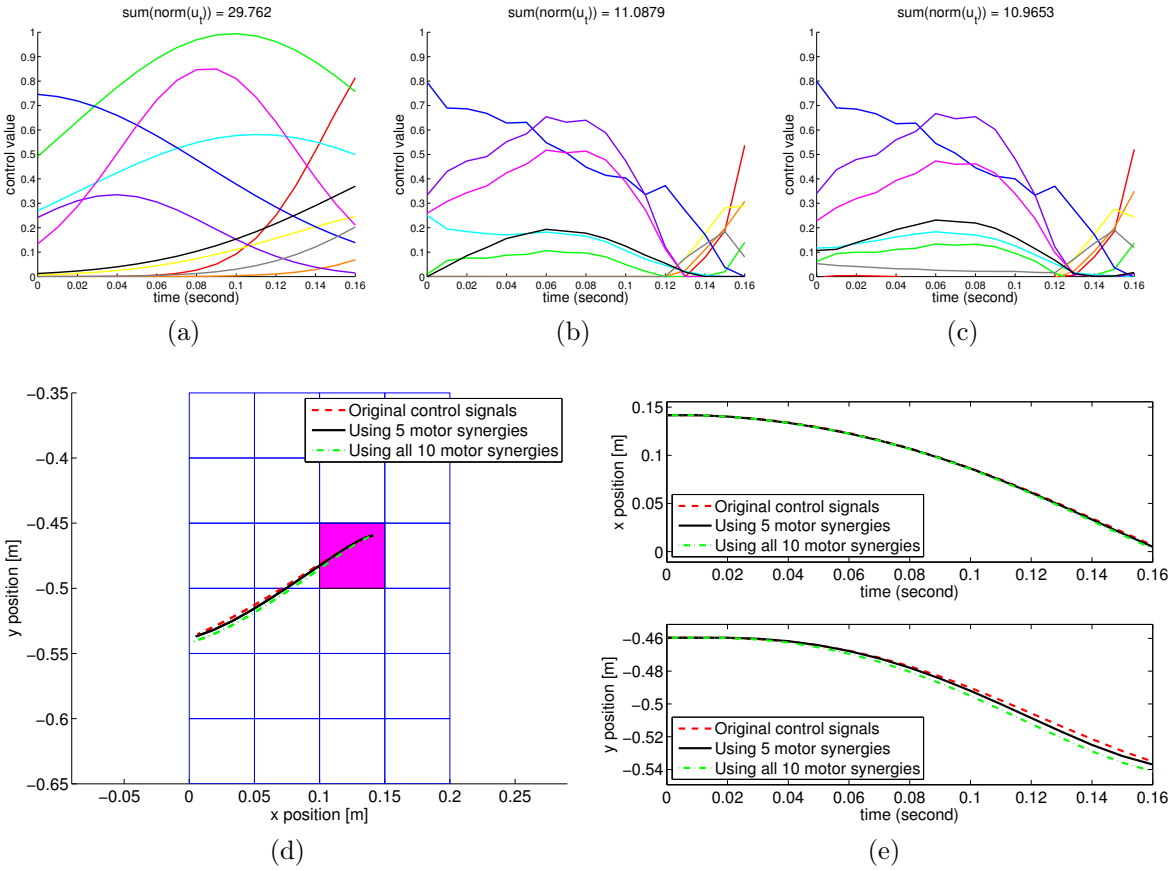


Figure 4-4: The regeneration of a selected reaching-like movement according to the end-effector efficiency. (a) The original randomly initialized parameterized control signals. (b): Control signals reconstructed using all the extracted 10 muscle synergies. These signals were the estimated optimal control signals. (c): Control signals reconstructed using the first 5 muscle synergies. These signals were used for the regeneration. (d): The end-effector trajectories in the task space generated by the control signals in Fig. 4-4a, Fig. 4-4b and Fig. 4-4c are shown by the dashed line, the solid line and the dash-dot line, respectively. The grid indicates a rectangular area and the regions where the training data were collected, for the manipulation task in section 4.7. This trajectory starts from the highlighted region. (e): The histories of the x and y position. These graphs show that the reaching-like movements could be reproduced by using low-dimensional control signals via muscle synergies.

plying the same sequence of the extracted low-dimensional control signals with the same initial conditions. Fig. 4-4 depicts the difference before and after a regeneration of a movement using low-dimensional control signals. It was observed that the low-dimensional control signals produced similar movements, with smaller control effort. The trajectories generated by muscle synergies deviated from the original one because of accumulative difference between the original control signals (Fig. 4-4a) and the reconstructed signals from muscle synergies. The control signals reconstructed from fewer muscle synergies (Fig. 4-4c) were almost the same as the ones from all synergies (Fig. 4-4b) and thus produced almost the same trajectories. The regeneration process does not have much influence to the distribution of the selected movement data, and thus does not affect the estimation of inverse dynamics and control performance.

4.6.3 Overall procedure

The procedure for estimating the inverse dynamics with muscle synergies is summarized in Table 4.1. In Step 1, and in Step 5, the forward dynamics and the low-dimensional inverse dynamics are estimated independently for each dimension of $\ddot{\boldsymbol{y}}$ and $\boldsymbol{a}$, respectively. In Step 2, since the estimates $\tilde{\alpha}$, and $\tilde{\beta}$ (not the true α and β) are used in solving (4.14), there may exist $\|\boldsymbol{u}_i^{min}\|_2 \geq \|\boldsymbol{u}_i\|_2$. In this case, the optimal control signal was set to the original control signal $\boldsymbol{u}_i^{min} = \boldsymbol{u}_i$ in implementation. In Step 4, the regeneration of movements ensures that each $\ddot{\boldsymbol{y}}_i^a$ is the outcome of applying $\boldsymbol{a}_i$ at $\tilde{\boldsymbol{x}}_i^a$. The application of PCA on $\boldsymbol{u}_i^{min}$ also acts as a whitening step to decorrelate the data in Step 5. The estimation error of $\boldsymbol{u}_i^{min}$ and the possible existence of local minima when solving the quadratic programming problem may affect the estimation of the inverse dynamics. However, the evaluation in the next section shows that the estimated inverse dynamics mapping could still be applied to achieve a tracking control task.

The procedure can also be interpreted as a self-learning process of extracting muscle synergies for low-dimensional control. The extraction of muscle synergies corresponds to the extraction of control skills from random movements (Step 1 to Step 3). Then, the extracted control skills are practiced by replicating the same

Table 4.1: Procedure for estimation of inverse dynamics for low-dimensional control. This scheme can be considered as a realization of a simple self-learning process of muscle synergies for low-dimensional control.

Given N training samples $\{\boldsymbol{u}_i, \tilde{\boldsymbol{x}}_i, \ddot{\boldsymbol{y}}\}_{i=1}^{N}$ of Q movements

Extracting muscle synergies (control skills)

Step 1: Estimate $\tilde{\alpha}(\tilde{\boldsymbol{x}})$ and $\tilde{\beta}(\tilde{\boldsymbol{x}})$ from the forward dynamics:
$\ddot{\boldsymbol{y}} = \alpha(\tilde{\boldsymbol{x}}) + \beta(\tilde{\boldsymbol{x}})\boldsymbol{u}$

Step 2: Estimate the optimal control signals using $\tilde{\alpha}(\tilde{\boldsymbol{x}})$ and $\tilde{\beta}(\tilde{\boldsymbol{x}})$ by solving (4.14).

Step 3: Extract muscle synergies from $\boldsymbol{u}_i^{min}$ using PCA:
$\boldsymbol{u}_i^{min} \approx \mathbf{W}\boldsymbol{a}_i + \bar{\boldsymbol{u}}$

Regeneration of movements (Practice with the control skills)

Step 4: Re-generate the Q movements by using the corresponding lower dimensional control sequence $\boldsymbol{a}_i$ with the same initial condition. Denotes the regenerated samples as $(\boldsymbol{a}_i, \tilde{\boldsymbol{x}}_i^a, \ddot{\boldsymbol{y}}_i^a)$

Estimation of inverse dynamics (Remember the uses of the control skill)

Step 5: Use the regenerated samples $(\boldsymbol{a}_i, \tilde{\boldsymbol{x}}_i^a, \ddot{\boldsymbol{y}}_i^a)$ for estimation of the inverse dynamics:
$\boldsymbol{a} = \beta^{\dagger}(\tilde{\boldsymbol{x}}^a)(\ddot{\boldsymbol{y}}^a - \alpha(\tilde{\boldsymbol{x}}^a))$

---

movements (Step 4). Finally, the utilization of the extracted control skills are remembered (Step 5).

In simulation, it is found that if the sparsification technique in described (4.11) and (4.12) was used instead of solving equation (4.10), it might remove important data samples, resulting in bad control performance. It can be solved by keeping a portion of training data that always contribute in the estimation. This can be accomplished easily by randomly selecting a portion of training data in each region in the workspace, and setting the selected portions of all the regions as the "free bases" in the SparseBayes 2.0 [139]. It does not remove "free bases" from the training data during sparsification.

4.7 Experiments

4.7.1 Simulation setup

The planar 3-DOF human-like robotic arm actuated by 10 muscles described in Appendix A was used as a simulation platform. The control task was to follow a figure of "8" trajectory for four cycles locating within a rectangular area (0.2m X 0.3m) with lower left corner at (0,-0.65) in the Cartesian plane. The rectangular area was divided into 24 square regions (0.05m X 0.05m). The figure of "8" trajectory consists of all movement directions and thus suitable for the evaluation of the extracted muscle synergies for general manipulation task. In data collection, 192 of 5000 random reaching-like movements were collected from the rectangular area, according to the end-effector efficiency as described in section 4.2.1. (The best 2 movements were chosen for each of the four moving directions). The total number of samples was 3774. (Samples outside the grids were excluded). All the samples were used for the estimation of the forward dynamics. For the estimation of the inverse dynamics, 1920 samples were used for faster computation. The samples were collected by randomly selecting 80 samples in each region. In order to examine the robustness to measurement noise, $\boldsymbol{x}$, $\dot{\boldsymbol{x}}$, $\boldsymbol{y}$, $\dot{\boldsymbol{y}}$ and $\ddot{\boldsymbol{y}}$ were corrupted with Gaussian noise. The signals were

corrupted as $\tilde{\boldsymbol{s}}' = \boldsymbol{s}' + \hat{\sigma}\boldsymbol{\zeta}$ where $\boldsymbol{\zeta}$ was drawn from the Gaussian distribution with zero mean and identity covariance, where $\hat{\sigma} = 0.2^o$, $\hat{\sigma} = 2^o\text{s}^{-1}$ and $\hat{\sigma} = 2^o\text{s}^{-2}$ for $\boldsymbol{x}$, $\dot{\boldsymbol{x}}$ and $\ddot{\boldsymbol{x}}$, respectively; $\hat{\sigma} = 0.002\text{m}$, $\hat{\sigma} = 0.02\text{ms}^{-1}$ and $\hat{\sigma} = 0.2\text{ms}^{-2}$ for $\boldsymbol{y}$, $\dot{\boldsymbol{y}}$ and $\ddot{\boldsymbol{y}}$, respectively. The noisy measurements were used for estimations of the dynamics and during control.

The simulation was carried out following the steps described in Table 4.1. Both the forward dynamics and the inverse dynamics were estimated with the Gaussian kernel

$$k(\tilde{\boldsymbol{x}}_p, \tilde{\boldsymbol{x}}_q) = \exp\left(-\frac{1}{2}(\tilde{\boldsymbol{x}}_p - \tilde{\boldsymbol{x}}_q)^T\mathbf{L}(\tilde{\boldsymbol{x}}_p - \tilde{\boldsymbol{x}}_q)\right). \tag{4.29}$$

$\mathbf{L}$ was a diagonal matrix with entries $L_m = \sigma^{-1}(\tilde{x}_m^{max} - \tilde{x}_m^{min})^{-2}$ where $\tilde{x}_m^{max}$ and $\tilde{x}_m^{min}$ were the maximum and minimum of the mth dimension in the training data. σ was the parameter of the kernel width. σ and the regularization constant γ were determined by cross validation. Isotropic σ and γ were used for simplicity. The importance of each datum was assumed to be the same and thus $\mathbf{Q} = \mathbf{I}$ was set. The performance of the sparsification technique as described in section 4.3 was also evaluated, where 20% of the 1920 samples were set to be always incorporated in the inverse dynamics estimation. (set as "free bases" in SparseBayes 2.0). The muscle synergies were extracted from the optimal control signals $\boldsymbol{u}_i^{min}$ by PCA using the MATLAB function *pca*. The 6 most important muscle synergies are depicted in Fig. 4-5. All the simulations were carried out in MATLAB.

4.7.2 Tracking a figure of 8 trajectory

The control parameters were set to $\lambda = 10$, $\boldsymbol{k}_s = [10, 10]^T$, $\mathbf{K}_{p0} = 30\mathbf{I}$, $\mathbf{K}_{v0} = \sqrt{30}\mathbf{I}$. Fig. 4-6, Fig. 4-7, and Fig. 4-8 show the results of tracking the desired trajectory for four cycles, in which 5 muscle synergies were utilized. The robot successfully followed the desired trajectory using low control gain $\boldsymbol{k}_s$. The null space control (4.25) and (4.26) were able to keep the joints away from the joint limits. It should be noticed that the muscle synergies were extracted from data of movements different from the desired trajectory.

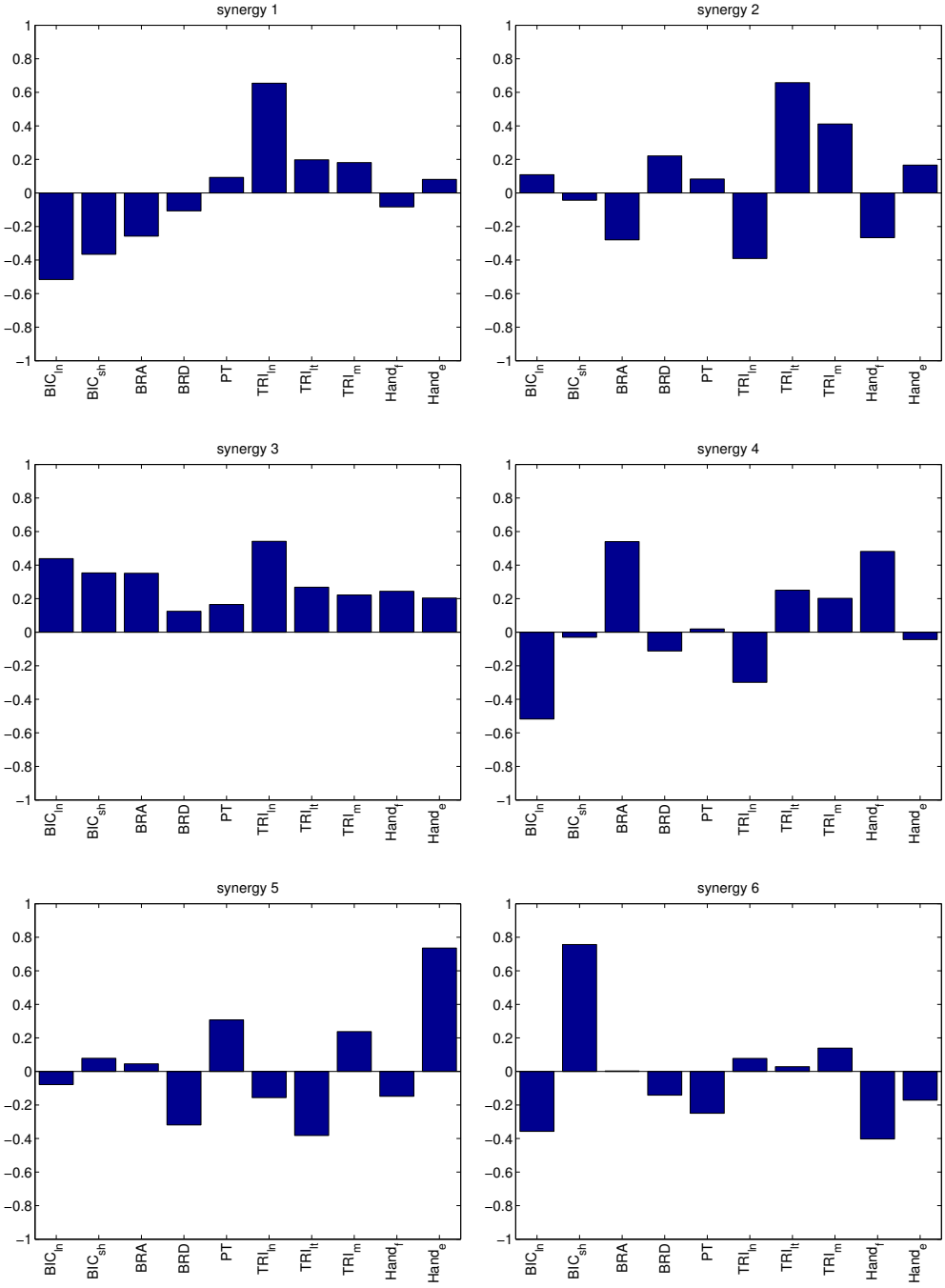


Figure 4-5: 6 of the 10 muscle synergies extracted from the estimated optimal control signals. Each synergy specifies co-activations of the 10 muscles of the robot arm. The low dimensional control signals activate these synergies to control all the muscles.

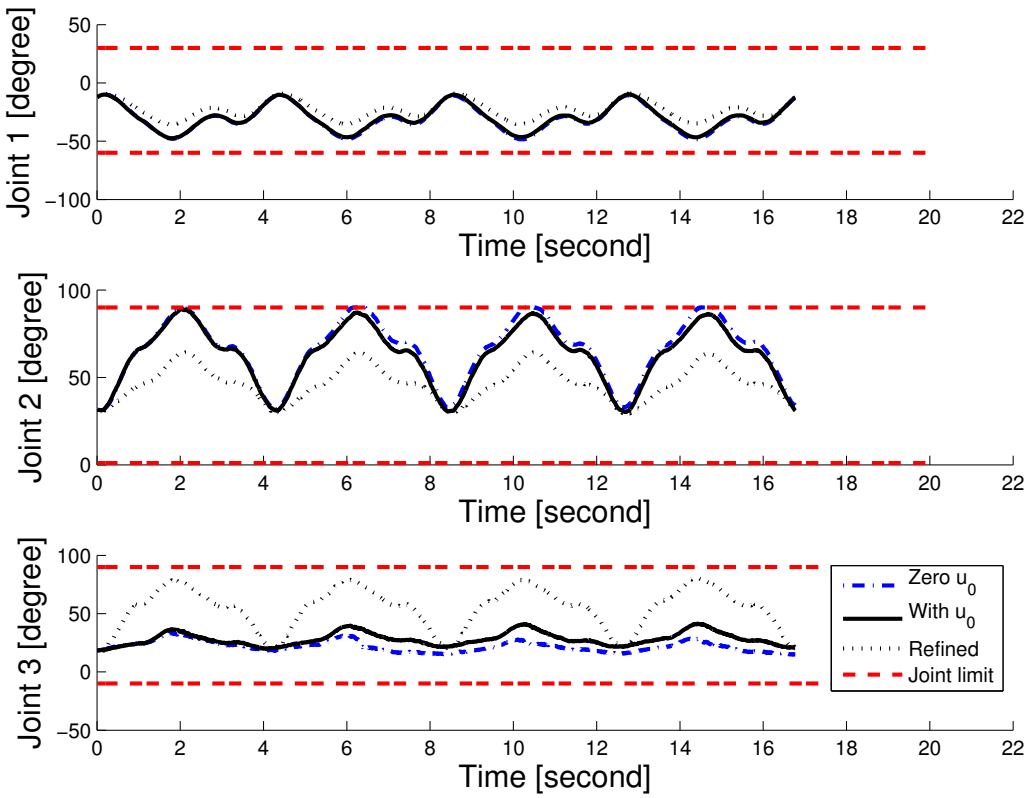


Figure 4-6: Trajectory history in joint space. In the case of using zero null space control $\boldsymbol{u}_0 = 0$, the robot collided at the upper limit of joint 2 at 90^o. With the aid of null space control, the robot stayed within the joint limits without collisions. The collisions cause the deviation from the desired trajectory depicted in Fig. 5-6b and Fig. 5-6d. The dot lines are the results utilizing refined muscle synergies. These results demonstrate the ability of the null space control $\boldsymbol{u}_0$ that stabilizing the robot in the joint space and perform secondary goal.

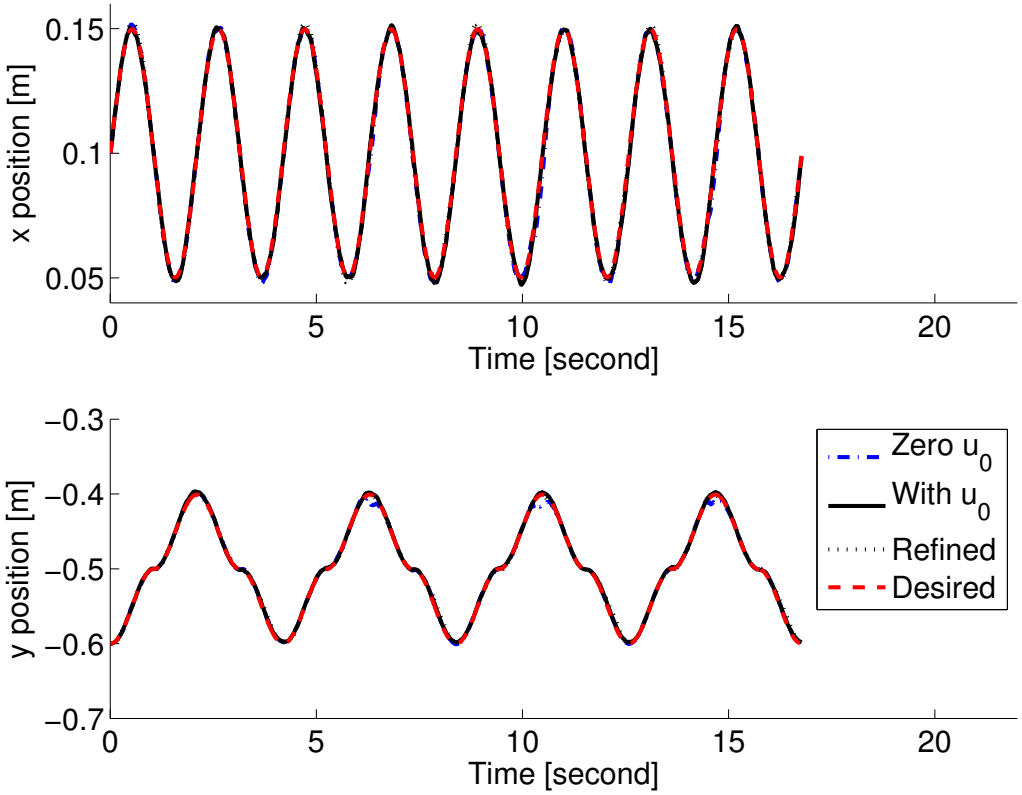


Figure 4-7: Trajectory history in the task space. In the case without null space control $\boldsymbol{u}_0 = 0$, the robot moved away from the desired trajectory in the y position, due to the collisions at the joint limit as depicted in Fig. 5-6c. The dot lines are the results utilizing refined muscle synergies.

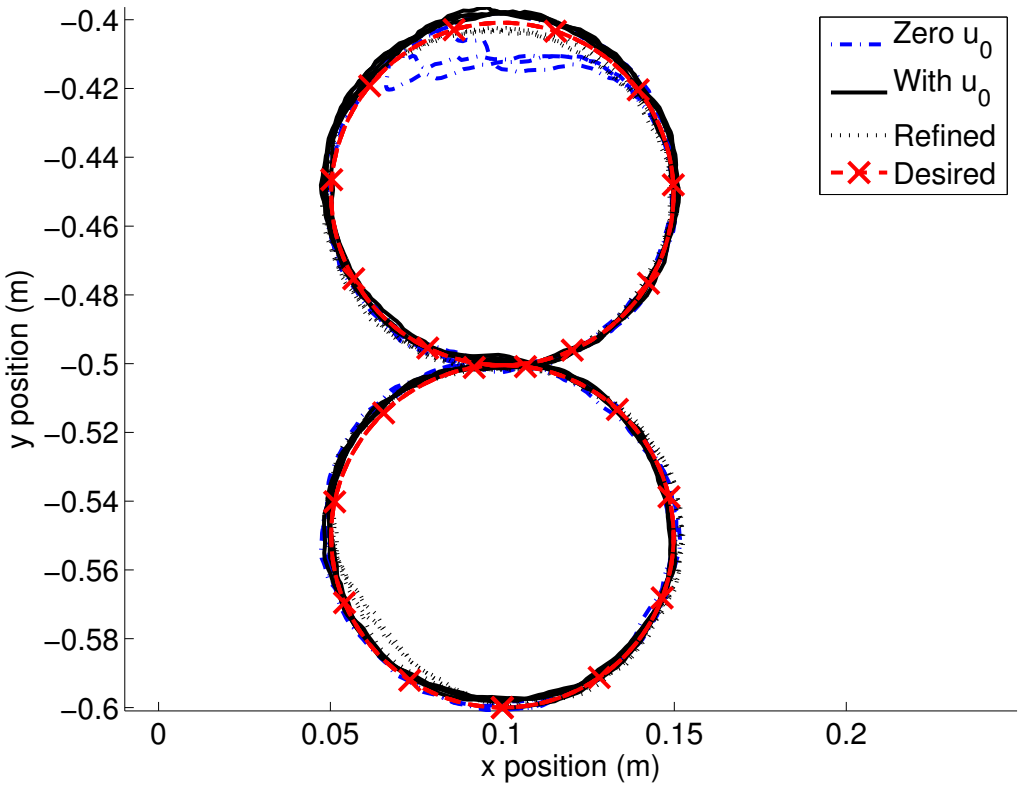


Figure 4-8: Tracking a figure of "8" trajectory for four cycles. The dot line is the result utilizing refined muscle synergies.

Table 4.2: Control performance vs Number of muscle synergies utilized.

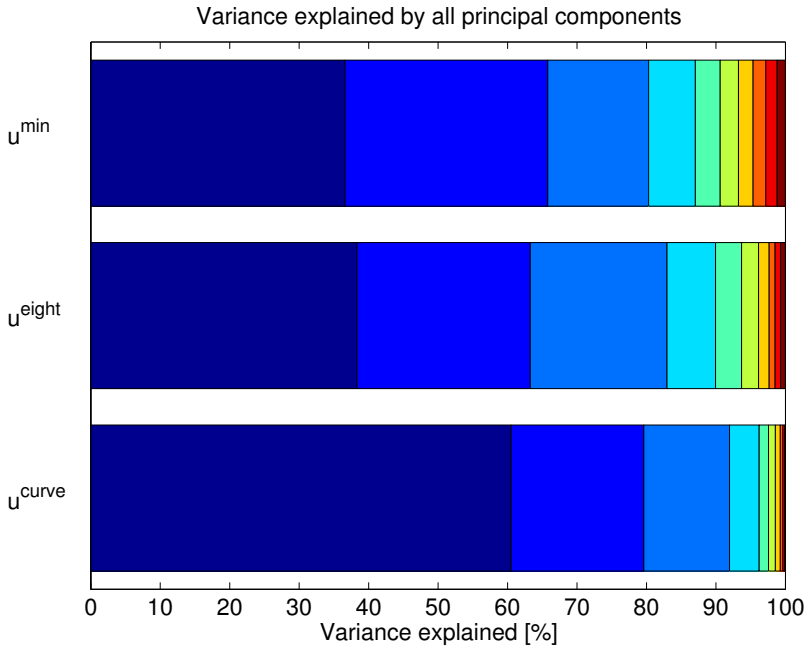
		Average position error from the desired trajectory [$\times 10^{-6}$m]		
Number of synergies	Total variance explained	All training data	Sparsification	
2	65.8%	107.25 (88.0%*)	7010 (88.2%*)	20.5%
3	80.3%	77.91 (91.3%*)	1654 (95.9%*)	25.8%
4	87.0%	74.61 (98.5%)	1252 (100%*)	25.3%
5	90.6%	1.85 (100%)	2.18 (100%)	31.0%
6	93.2%	3.04 (100%)	3.59 (100%)	22.8%
7	95.3%	2.35 (100%)	3.56 (100%)	25.2%
8	97.2%	2.39 (100%)	4.93 (100%)	39.3%
9	98.8%	5.71 (100%)	4.94 (100%)	42.7%
10	100%	9.80 (100%)	12.77 (100%)	43.0%

The values in the brackets are the average percentage of following one cycles of the figure of "8" trajectory, before the instantaneous position error > 0.03m.
The asterisks indicate collisions occurred at joint limits.
The last column is the percentage of the training data remained after sparsification.

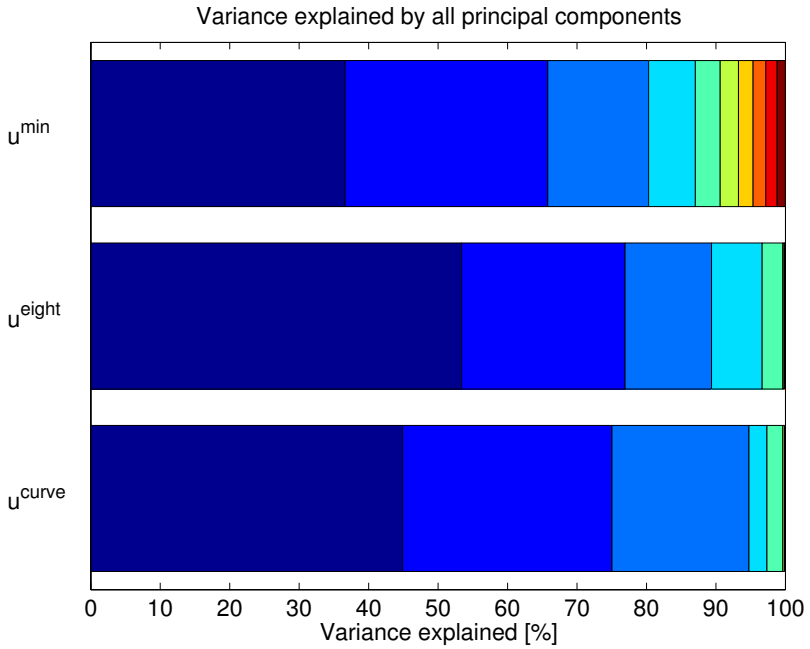
Table 4.2 summarizes the control performance when a different number of muscle synergies were utilized. The position errors were the average of 10 runs of following the figure of "8" trajectory for four cycles in each case. The third column shows the results when inverse dynamics was estimated by all training data (solving equations (4.10)), whereas the fourth and the fifth columns are the results when sparsification was employed. The robot failed to follow the desired trajectory when 4 or fewer muscle synergies were utilized, and succeed when more than 4 muscle synergies were utilized. When sparsification was employed, good control accuracy could also be achieved when 5 or more muscle synergies were utilized. It was also observed that after sparsification, about 20% to 30% of the training data remained when 7 or fewer muscle synergies were utilized, whereas about 40% of the training data remained when 8 or more muscle synergies were utilized.

4.7.3 Dimension reduction on controlled movements and muscle synergies refinement

It was expected that the degree of dimension reduction was better than that listed in the 2nd column in Table 4.2, when a well-controlled movement data was the data source. This subsection investigates how many muscle synergies were needed to draw the figure of "8" trajectory, by performing PCA on the control signals of the tracking result when all 10 muscle synergies were utilized. Fig. 4-9 compares the variance characteristics between the tracking results and the estimated optimal control signals. Fig. 4-9a indicates that 5 principal components were needed to explain over 90% of the total variance of $\boldsymbol{u}^{eight}$. This is consistent with the control performance in Table 4.2. For simpler movement of drawing the first quarter of the figure of "8"($\boldsymbol{u}^{curve}$), better dimension reduction performance could be achieved. In Fig. 4-9b, over 99% of the total variance of $\boldsymbol{u}^{eight}$ could be explained by 5 PCs, because the control signals $\boldsymbol{u}^{eight}$ was computed by utilizing 5 muscle synergies and thus had a better degree of dimension reduction. These results motivated us to further refine the muscle synergies to reduce the number of muscle synergies required.



(a) Controlled with 10 synergies



(b) Controlled with 5 synergies

Figure 4-9: Percentage of variance explained by all the principal components of PCA, which was performed on the control signals of (top) $\boldsymbol{u}^{min}$: the estimated optimal control signals of the reaching-like movements selected according to the end-effector efficiency. It is the same as the data in the second column of Table 4.2; (2^{nd} row) $\boldsymbol{u}^{eight}$: the control signals drawing the figure of "8"; and (bottom) $\boldsymbol{u}^{curve}$: the first quarter of $\boldsymbol{u}^{eight}$ drawing the lower right half circle of the figure of "8".

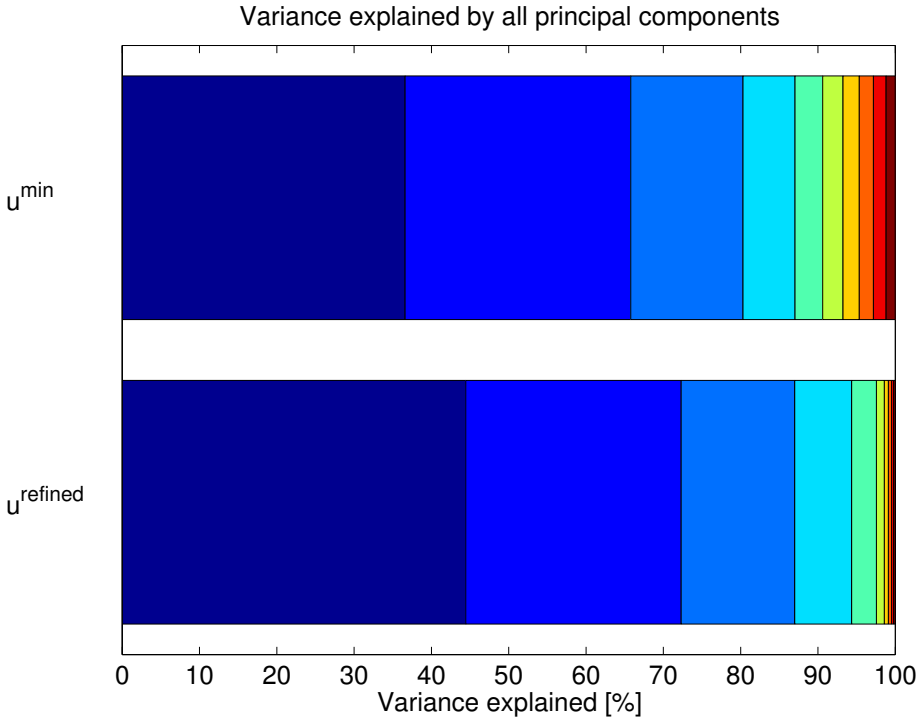


Figure 4-10: Percentage of variance explained by all the principal components of PCA, which was performed on the control signals of (top) $\boldsymbol{u}^{min}$: the estimated optimal control signals of the reaching-like movements selected according to the end-effector efficiency. It is the same as the data in the second column of Table 4.2; (2^{nd} row) $\boldsymbol{u}^{refined}$: the control signals of the 192 selected reaching-like movements tracked by the sliding controller with 5 synergies.

It was attempted to refine the muscle synergies in order to further reduce the number of synergies needed. After learning the inverse dynamics, the 192 selected reaching-like movements were tracked utilizing 5 muscle synergies, with the same initial conditions. The samples of these 192 tracked movements were used as the source to extract muscle synergies, to regenerate movements and to learn the inverse dynamics following the procedure in Table 4.1. In this case, because the control signals for synergy extraction were constituted by 5 synergies, ideally performing PCA would give the first 5 principal components explaining 100% of the total variance. Fig. 4-10 depicts the variance characteristics of the refined muscle synergies, in which the first 5 principal components explain 98% of the total variance. This is possibly because the control signals were bounded within 0 and $\boldsymbol{u}^{ub}$; This bounding process can be considered as adding noise to the control signals, which contributes the remaining 2% variance. The figure of "8" could be tracked utilizing 4 muscle synergies without collision to the joint limits, with average position error 5.15 $\times 10^{-6}$m (10trials). The tracking trajectories are depicted in Fig. 4-6, Fig. 4-7 and Fig. 4-8.

4.8 Discussion

4.8.1 The minimum number of synergies required

Compared with chapter 3, this chapter allows positive and negative synergy activations. Therefore, it is possible that such bidirectional activation of a synergy may produce 2 distinct possible motion directions, thereby there may exist 2 synergies that may produce 4 directions for producing motion in all possible directions, as illustrated in the Fig 4-11.

However, according to the result in Table 4.2, 5 synergies were required to track the desired trajectory with small tracking error. One possible reason is that the moving direction produced by a synergy depends on the robot posture. According to (4.1), the end-effector acceleration of the robot is given by multiplying the muscle activation $\boldsymbol{u}$ by a nonlinear function $\beta(\tilde{\boldsymbol{x}})$ where $\tilde{\boldsymbol{x}}$ consists of joint configurations

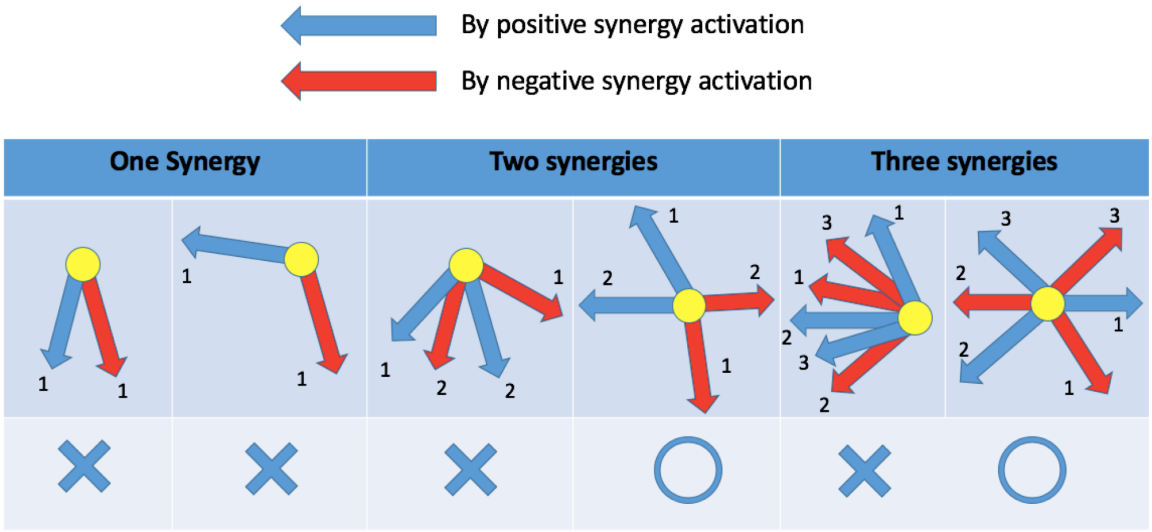


Figure 4-11: Possible moving directions of the robot end-effector when 3 muscle synergies, $\boldsymbol{w}_1$, $\boldsymbol{w}_2$ and $\boldsymbol{w}_3$ are activated individually. The circles denote the end-effector positions on a 2D task space, and each arrow denotes a moving direction when a muscle synergy is activated. The numbers "1", "2" and "3" denote the indexes of the synergies $\boldsymbol{w}_1$, $\boldsymbol{w}_2$ and $\boldsymbol{w}_3$, respectively. This figure illustrates that activating a synergy may move the end-effector in two distinct directions (the second left) since both positive synergy activation and negative synergy activation is allowed; At least 2 synergies may be required to move the end-effector in all possible directions on a 2D horizontal task space.

$\boldsymbol{x}$ and joint velocities $\dot{\boldsymbol{x}}$, followed by adding another nonlinear function $\alpha(\tilde{\boldsymbol{x}})$, which must be a nonzero term because it is the passive dynamics of the robot in task space. Moreover, equation (4.1) also implies that activating a synergy with opposite sign does not necessarily produce opposite moving direction, because of the presence of the nonzero term $\alpha(\tilde{\boldsymbol{x}})$.

4.8.2 Determining the best number of synergies

For the experiment 4.7.2, the best number of synergies compromising the control complexity (dimensionality) and the control performance of the synergies-based tracking controller can be determined according to the control performance in the task space and the joint space. The control performance in the task space is measured according to the tracking error (the 3rd column in Table 4.2) of the end-effector, and the control performance in the joint space is measured according to the occur-

rence of collisions to the joint limits (indicated with an asterisk mark in Table 4.2 if collisions occurred). From Table 4.2, it is observed that the control accuracy of the end-effector in the task space improves when the number of synergies increased from 2 to 4, attained similar level of small tracking error (from 1.9×10^{-6}m to 3.0×10^{-6}m) when 5 to 8 synergies were utilized, and slightly worsened to larger error (smaller than 10.0×10^{-6}m) when 9 and 10 synergies were utilized. The similar values of the tracking error for the cases of 5 or more synergies in Table 4.2 implies that set of the muscle activations to track the desired trajectory in Fig. 4-8 lies in a lower-dimensional control space. The synergies-based controller has a secondary control task goal, where the joints are regulated by the null space control term (the last term in (4.23), (4.25) and (4.26)) such that the joints are kept away from the limits without much interfering the control performance in the task space. The robot collided with the joint limits when the number of synergies utilized was smaller than 4 (or 5 when the sparsification technique was used), indicating by the asterisks next to the tracking error in Table 4.2. Considering the best number of synergies as the minimum number of synergies that can achieve good control performance in both the task space and the joint space. According to Table 4.2, the best number of synergies in experiment 4.7.2 can be determined as 5.

Note that the control performances were assessed for the particular tracking control task in experiment 4.7.2, it may need a different number of synergies to accomplish different desired task space trajectories. Moreover, the assessment of the control performance in the joint space can only indicate whether the controller can keep the joints away from the limits. Further investigations of the control performance to achieve a variety of control tasks using different desired task space trajectories, and the study of relationship between the number of synergies and the control performance in the joint space are the future works.

4.9 Summary

In this chapter, the problem of extracting muscle synergies and its application to a task space controller for a musculoskeletal robot were studied. It is proposed to estimate the optimal muscle activations in the sense of minimum control effort, using robot's forward dynamics estimated by a system identification technique. It is shown that when extracting muscle synergies from the estimated optimal muscle activations, dimensionality reduction can be achieved. It is also demonstrated that using the estimated optimal data as the training source has the advantages of enabling direct estimation of the generally ill-posed robot's inverse dynamics, and providing maximum capacity for null space control within the admissible control range. A sliding control law for a class of nonlinear overactuated systems, as well as a kernel-based formulation for estimating the robot's forward and inverse dynamics, are provided. Numerical simulation results show that the proposed algorithm can extract muscle synergies for general end-effector manipulation and achieve low-dimensional control. A sparsification technique is proposed to handle large data set. It is also shown that the muscle synergies required can be further reduced, by extracting muscle synergies from reaching-like movements where the robot is controlled in a reduced dimensionality by the tracking controller with muscle synergies.

Chapter 5

Obtaining muscle synergies in a goal-directed exploration scheme

Although chapter 4 has developed a technique for extracting muscle synergies, it is assumed that the robot can start the exploration from an arbitrary resting configuration; Such assumption may not be always satisfied. In this chapter, a data collection method is proposed based on a goal-directed exploration strategy, where the robot explores unknown task space area by trying to reach designated targets spreading over the task space successively, without resetting to a resting configuration. During exploration, muscle synergies are obtained from local data sample, aiding in the estimation of the robot's inverse dynamics in reduced control dimensionality. The robot is controlled by a task space controller with null space control that can regulate the joint away from joint limits, using the inverse dynamics obtained from local data. In simulation evaluation, the proposed methods enabled a human-like robotic arm to collect data sample of omnidirectional reaching movements in an exploration task of a 2D task space. It was demonstrated that the collected data sample could be used a training source to extract a set of muscle synergies and to establish a controller for manipulating the robot in reduced control dimensionality. Two bio-inspired strategies, goal-directed exploration and muscle synergies, which are is highly plausible in human beings, are integrated into engineering control techniques of musculoskeletal robots. The proposed method enables a robot to obtain muscle synergies by itself, which is a

step towards the development of autonomous musculoskeletal robots.

5.1 Introduction

Despite musculoskeletal robots have flexible and compliant structures that potentially provide advantages such as dexterity and safety, control difficulties of such complicated structure hinders development for real applications. Musculoskeletal robots usually have many joints and actuators, so that it can achieve a task in many different combinations of joint configurations and muscle activations. The general problem of control the robot is about how to coordinate so many actuators to achieve a task such as reaching an object or following a trajectory. Biological studies of human motor control might give us inspirations.

The concept of muscle synergies, a hypothesis to explain how human control their bodies, can reduce control complexity of musculoskeletal robots. It has been suggested that humans coordinate groups of muscle co-activations called muscle synergies, instead of controlling muscles independently [13, 117, 119]. In many engineering applications, the concept of muscles synergies was adopted for the purpose of reducing dimensionality. For example, muscle synergies acted in transferring human's motion from high dimensional electromyography (EMG) signals to a human-like low dimensional robot motion in a real-time tele-operation application [9]. Muscle synergies has also been employed in solving computationally expensive problems such as reinforcement learning of reaching movements of a full-torso, simulated musculoskeletal robot [128], or point-to-point manipulation of a redundant six-muscle simulated robotic arm using optimal control theory [96].

Muscle synergies can be extracted from control signals of a given data sample with inherent statistical regularities [119, 146], using pattern recognition tools. Examples of such data sample are electromyography (EMG) signals of specific movements [24, 110, 147] and optimized control signals with respect to certain criteria [96, 148, 149]. In chapter 4, a method to extract muscle synergies from data sample of randomly parameterized control signals which do not possess any statistical regularities is proposed.

However, not only statistical regularities of the control signals, but also the corresponding joint space and task space data in the sample have to be considered. For instance, muscle synergies extracted from control signals of uni-directional movements are unlikely be utilized for producing multi-directional movements. For manipulation within the robot task space to perform various tasks, data sample of omnidirectional movements is an appropriate choice.

Goal-directed exploration, an effective motor skill learning strategy observed in the study of human motor learning, is a promising approach to enable robots obtaining muscle synergies. Goal-directed exploration has been observed in motor skill development in human [150, 151], where infants obtain motor skills by doing goal-directed actions, such as trying to reach an object even they will fail [152]. Especially in applications of high-dimensional and redundant robotic systems, goal-directed strategy achieves faster exploration using much fewer sample points [153], based on directly and actively exploring often low-dimensional task space instead of often high-dimensional joint space or control space [137, 154]. It has been shown that goal-directed exploration enabled efficient data collection in applications of obtaining inverse kinematics models for controlling a simulated high-dimensional and redundant robots [137] and a real bionic elephant trunk robot [153], where prior knowledge of the robots were not given. In [154], synergies of parameterized kinematics control inputs of a simulated quadruped robot were incorporated in an advanced goal-directed exploration scheme where the goals in task space were self-generated. However, the number of control variables was still larger than the number of degree-of-freedom (DOF) of the quadruped robot. Because the motion of a joint is often actuated redundantly by multiple muscles, kinematics control may not be adequate in musculoskeletal robots. In [155], a goal-directed exploration strategy was adopted in controlling a real, human-like musculoskeletal robotic arm actuated by many pneumatic muscles, but the inverse dynamics was estimated in full dimension. Currently, there is no example of obtaining muscle synergies by goal-directed exploration for controlling musculoskeletal robots.

This chapter proposes a method to extract muscle synergies by goal-directed exploration, and simultaneously obtain an inverse dynamics mapping for control without prior knowledge of the robot. The proposed method has three main features. First, in order to construct a controller with adequate manipulation ability using muscle synergies and inverse dynamics, the "goals"(targets) are spread within the task space in order to collect data sample of omnidirectional point-to-point movements. Second, to facilitate the exploration, muscle synergies are incorporated during exploration, such that the robot is controlled in reduced control dimensionality using inverse dynamics mapping updated from local data sample. Third, a task space feedback controller associated with null space control is employed to proceed exploration by effectively manipulating the end effector and regulating the joints from joint limits. Integration of the two bio-inspired strategies, goal-directed exploration and muscle synergies, is highly plausible in human beings, but few examples have been shown in literature of the integration of these strategies for the control of musculoskeletal robots. The proposed method enables robots to obtain muscle synergies by itself, which is an important step towards development of autonomous musculoskeletal robots.

5.2 The exploration scheme

A data sample having control signals that generate end effector movements with different moving directions covering the whole task space is more suitable for synergy extraction. For example, synergies extracted from a data sample of horizontal point-to-point movements from one specific location to one specific target in a 2D task space are unlikely to be utilized for producing vertical point-to-point movements. If a robot can start from rest anywhere in the whole task space, a data sample of end effector movements with different moving directions can be easily obtained, by randomly actuating the robot from rest at random locations in the task space. However, this "starting from anywhere" assumption may not be easily satisfied in reality.

This section presents an exploration scheme that enables a robot generates sample points of end effector movements with different moving directions for extraction of syn-

ergies. Section 5.2.1 first defines some notations for the ease of description. Then the overview of the proposed exploration scheme is given in section 5.2.2. Section 5.2.3, section 5.2.4 and section 5.2.5 give detailed description of three components: the initialization process, the try-to-reach process and the stable point positioning process, respectively.

5.2.1 Notation definitions

This chapter considers the same model of musculoskeletal robots as described in section 4.2 (copy from (4.1)):

$$\begin{aligned} \ddot{\boldsymbol{x}}(t) &= f(\tilde{\boldsymbol{x}}(t)) + g(\tilde{\boldsymbol{x}}(t))\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= h(\boldsymbol{x}(t)) \\ \ddot{\boldsymbol{y}}(t) &= \alpha(\tilde{\boldsymbol{x}}(t)) + \beta(\tilde{\boldsymbol{x}}(t))\boldsymbol{u}(t) \end{aligned} \tag{5.1}$$

where $\boldsymbol{x} \in \Re^{n^x}$, $\dot{\boldsymbol{x}} \in \Re^{n^x}$ and $\ddot{\boldsymbol{x}} \in \Re^{n^x}$ are the joint angles, velocities and accelerations, respectively. $\boldsymbol{y} = [y_1, ..., y_{n^y}]^T \in \Re^{n^y}$, $\dot{\boldsymbol{y}} = [\dot{y}_1, ..., \dot{y}_{n^y}]^T \in \Re^{n^y}$ and $\ddot{\boldsymbol{y}} = [\ddot{y}_1, ..., \ddot{y}_{n^y}]^T \in \Re^{n^y}$ are is the position, velocity and acceleration of the end effector in the task space $\mathcal{S}$ of the robot, represented in the Cartesian coordinates. $\boldsymbol{u} \in \Re^{n^u}$ is the control input of actuators (muscles) with dimension $n^u \geq n^x$ bounded by $\mathbf{0} \leq \boldsymbol{u} \leq \boldsymbol{u}^{ub}$, The functions f, g, h, α and β are assumed to be unknown. Let $\tilde{\boldsymbol{x}} = [\boldsymbol{x}^T, \dot{\boldsymbol{x}}^T] \in \Re^{2n^x}, \tilde{\boldsymbol{x}} \in \mathcal{X}$ where $\mathcal{X}$ denotes the robot state space and T denotes the transpose operation. Also we denote $\boldsymbol{u} \in \mathcal{U}$, $\ddot{\boldsymbol{x}} \in \mathcal{X}^{acc}$, $\ddot{\boldsymbol{y}} \in \mathcal{Y}^{acc}$, where $\mathcal{U}$, $\mathcal{X}^{acc}$, $\mathcal{Y}^{acc}$ are the spaces of control, joint acceleration space and end effector acceleration, respectively. The following notations are also introduced for the ease of description:

- $\boldsymbol{x}_{home}$: A home configuration that the robot can always reset to.
- $\mathcal{S}$: The whole task space to be explored. This task space is divided into R smaller regions $\mathcal{S}_l, l = 0, ..., R-1$. The region $\mathcal{S}_0$ includes the home configuration $\boldsymbol{x}_{home}$.
- $\mathbf{Y}^{mn} = \{\boldsymbol{y}_k^*, \dot{\boldsymbol{y}}_k^*, \ddot{\boldsymbol{y}}_k^*, \boldsymbol{x}_k^*, \dot{\boldsymbol{x}}_k^*, \ddot{\boldsymbol{x}}_k^*\}_{k=1}^{\bar{N}}$: A set of vector series defining a desired tra-

jectory connecting region $\mathcal{S}_m$ and region $\mathcal{S}_n$. k denotes the time index such that time $t = (k-1)\Delta t$ where Δt is a fixed time step. Here k is used to emphasize that it is the time index. However, in the context, the symbol i is also used as sample index.

- $\{\mathbf{q}\}_{k=1}^{N} = \{\boldsymbol{a}_k, \boldsymbol{u}_k, \tilde{\boldsymbol{x}}_k, \ddot{\boldsymbol{x}}_k, \boldsymbol{y}_k, \dot{\boldsymbol{y}}_k, \ddot{\boldsymbol{y}}_k\}_{k=1}^{N}$: A data sample of a movement consists of N samples points. $\ddot{\boldsymbol{x}}_k$ and $\ddot{\boldsymbol{y}}_k$ are the resulting accelerations when applying $\boldsymbol{u}_k$ at $\tilde{\boldsymbol{x}}_k$. $\boldsymbol{a}_k$ is the corresponding synergy activation of $\boldsymbol{u}_k$ after extracting synergy. A sample point is denoted as $\mathbf{q}_k = \{\boldsymbol{a}_k, \boldsymbol{u}_k, \tilde{\boldsymbol{x}}_k, \ddot{\boldsymbol{x}}_k, \boldsymbol{y}_k, \dot{\boldsymbol{y}}_k, \ddot{\boldsymbol{y}}_k\}$.

- $\mathbf{D}$, $\mathbf{D}^{explored}$: $\mathbf{D}$ is used to denote a data sample consists of N movement sample points $\mathbf{D} = \{\mathbf{q}_i\}_{i=1}^{N}$. $\mathbf{D}^{explored}$ is specifically used to denote the library of collected sample points in exploration.

- $\mathbf{W}^{\mathbf{D}} \in \Re^{n^u \times M}$ and $\mathbf{W}_{\perp}^{\mathbf{D}} \in \Re^{n^u \times (n^u - M)}$: Synergies extracted from the control signals $\{\boldsymbol{u}_i\}_{i=1}^{N}$ in a data sample $\mathbf{D}$ using PCA:

$$\boldsymbol{u} \approx \sum_{j=1}^{M} \boldsymbol{w}_j a_j + \bar{\boldsymbol{u}} = \mathbf{W}^{\mathbf{D}} \boldsymbol{a} + \bar{\boldsymbol{u}}. \tag{5.2}$$

  $\mathbf{W}^{\mathbf{D}}$ consists of the first M important synergies (principal components) and $\mathbf{W}^{\mathbf{D}}$ consists of the remaining $n^u - M$ synergies.

- $\psi_{task}^{\mathbf{D}}$: The forward dynamics mappings in task space obtained from a data sample $\mathbf{D}$ which stores N sample points $\{\mathbf{q}_i\}_{i=1}^{N}$:

$$\psi_{task}^{\mathbf{D}} : \mathcal{X} \times \mathcal{U} \to \mathcal{Y}^{acc}, \quad (\tilde{\boldsymbol{x}}, \boldsymbol{u}) \mapsto \ddot{\boldsymbol{y}} = \alpha(\tilde{\boldsymbol{x}}) + \beta(\tilde{\boldsymbol{x}})\, \boldsymbol{u}. \tag{5.3}$$

- ${}^{\boldsymbol{a}}\zeta_{joint}^{\mathbf{D}}$, ${}^{\boldsymbol{a}}\zeta_{task}^{\mathbf{D}}$: Inverse dynamics mappings obtained from a data sample $\mathbf{D}$ which stores N sample points $\{\mathbf{q}_i\}_{i=1}^{N}$, for computing M-dimensional synergy activa-

tion $\boldsymbol{a} \in \Re^M$:

$$^{\boldsymbol{a}}\zeta^{\mathbf{D}}_{joint} : \mathcal{X} \times \mathcal{X}^{acc} \to \mathcal{A}, \qquad (\tilde{\boldsymbol{x}}, \ddot{\boldsymbol{x}}) \mapsto \boldsymbol{a} = \hat{g}^{\dagger}(\tilde{\boldsymbol{x}}) \left[\hat{f}(\tilde{\boldsymbol{x}}) - \ddot{\boldsymbol{x}}\right] \tag{5.4}$$

$$^{\boldsymbol{a}}\zeta^{\mathbf{D}}_{task} : \mathcal{X} \times \mathcal{Y}^{acc} \to \mathcal{A}, \qquad (\tilde{\boldsymbol{x}}, \ddot{\boldsymbol{y}}) \mapsto \boldsymbol{a} = \hat{\beta}^{\dagger}(\tilde{\boldsymbol{x}}) \left[\hat{\alpha}(\tilde{\boldsymbol{x}}) - \ddot{\boldsymbol{y}}\right]. \tag{5.5}$$

where $\hat{f}(\tilde{\boldsymbol{x}}_k)$, $\hat{g}(\tilde{\boldsymbol{x}}_k)$, $\hat{\alpha}(\tilde{\boldsymbol{x}}_k)$ and $\hat{\beta}(\tilde{\boldsymbol{x}}_k)$ can be formulated by substituting (5.2) into (5.1).

5.2.2 Overview

The exploration scheme is developed based on an assumption that the robot can reset its configuration to a home configuration $\boldsymbol{x}_{home}$. For example, the robot can return to a resting configuration under gravity by setting all actuation to zeros. It has been observed that infants reset posture in obtaining skills of reaching movement during goal-directed exploration [156]. Infants do not try to reach a target forever [137]. This assumption has also been made in other researches of goal-directed exploration [137, 153, 154].

The proposed scheme consists of three functional processes, the initialization process, the try-to-reach process and the stable point positioning process, in order to generate end effector movements with different moving directions. In the initialization process, the whole task space $\mathcal{S}$ is divided into smaller regions $\{\mathcal{S}_l\}_{l=0}^{R-1}$. In the region $\mathcal{S}_0$ where the home configuration $\boldsymbol{x}_{home}$ is included, a data sample $\mathbf{D}_0$ of movements generated by randomly parameterized control signals for the first try-to-reach process is collected. The exploration proceeds from $\mathcal{S}_0$, by iteratively executing the try-to-reach process and the stable point positioning process in order to explore all the regions $\{\mathcal{S}_l\}_{l=0}^{R-1}$. In a try-to-reach process, the robot attempts to move the end effector to reach the center of a region $\mathcal{S}_n$ that is in the neighborhood of the current starting region $\mathcal{S}_m$ where the robot starts the try-to-reach process, in a finite number of trials. In each trial, the robot moves from rest for a fixed number of time step, followed by resetting the end effector to the center of $\mathcal{S}_m$. The best trial in one try-to-reach process that is closest to the center of $\mathcal{S}_n$ at the last time step is

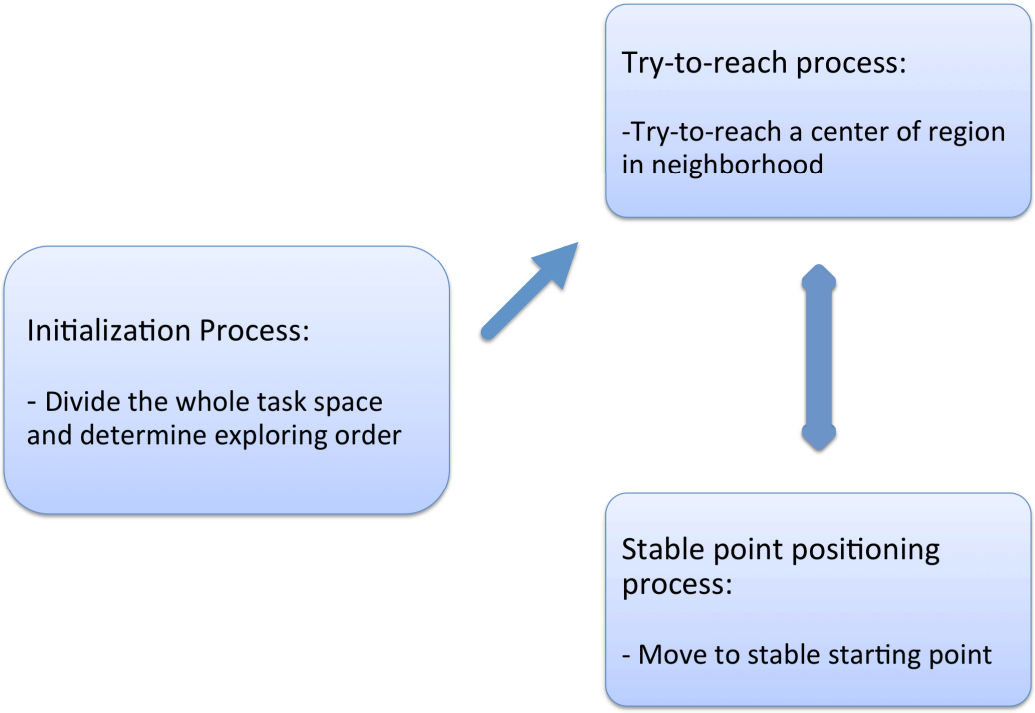
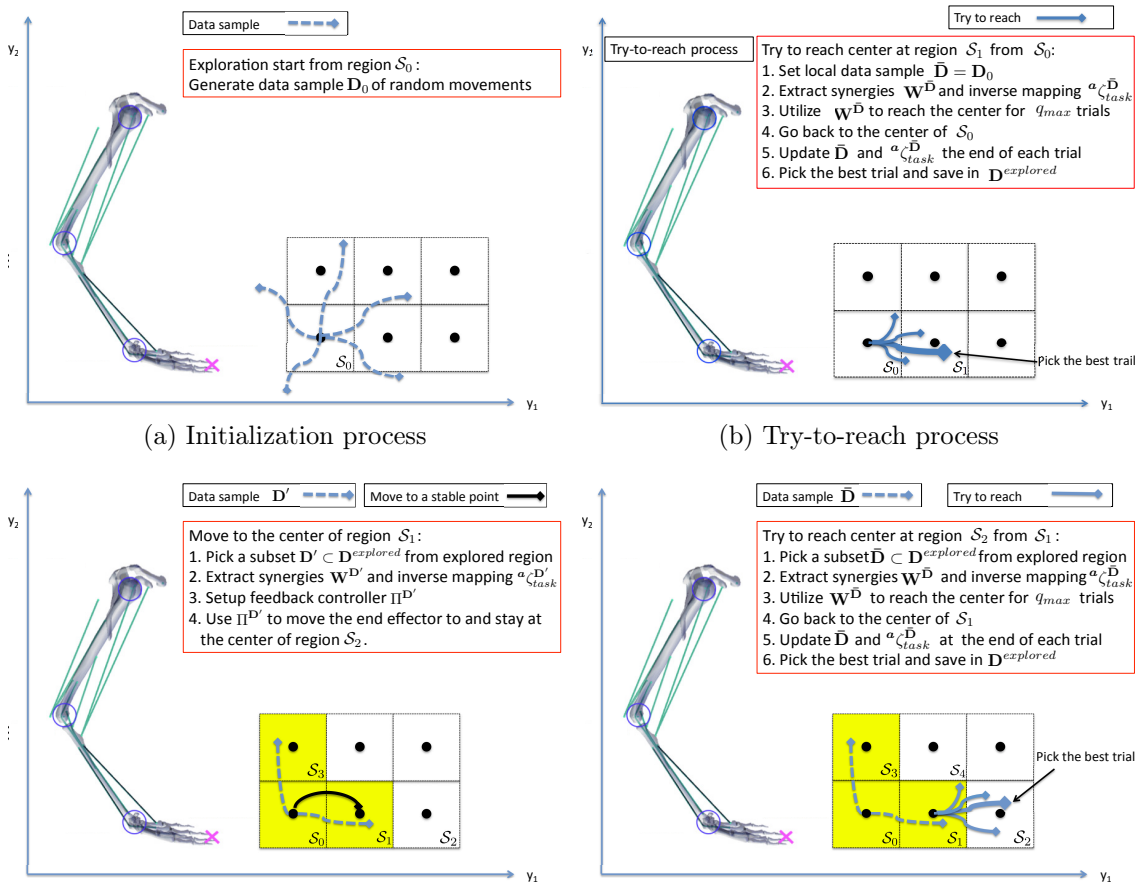


Figure 5-1: Conceptual flow of the proposed exploration scheme, which consists of three components. The task space being explored is divided into smaller regions with designated exploring order in the initialization process. The exploration proceeds by trying to reach designated region centers successively, in which the robot moves to a stable point in the task space in each trial of the point-to-point movement.

selected and whose sample points are collected in the library $\mathbf{D}^{explored}$. The robot continues the try-to-reach process until all designated regions $\mathcal{S}_n$ in the neighborhood of $\mathcal{S}_m$ has been "reached". Then, the robot proceeds to a new starting region $\mathcal{S}_{m'}$ adjacent to the current $\mathcal{S}_m$. The changing of starting region and the resetting of the end effector in each try-to-reach trial, are achieved by using a feedback task-space controller manipulating the robot to the center of $\mathcal{S}_{m'}$ and $\mathcal{S}_m$, respectively. Fig. 5-1 depicts the conceptual flow of the proposed exploration scheme. Fig. 5-2 illustrates the progression of exploration of a musculoskeletal robot in a 2D task space.

5.2.3 The initialization process

The division method is decisive to the data sample collected after exploration, since the robot explores the task space by moving from one region center to other



(a) Initialization process

(b) Try-to-reach process

(c) Stable point positioning process

(d) Try-to-reach process

Figure 5-2: An example of a human-like robotic arm exploring a 2D task space. (a) In the initialization process at the region $\mathcal{S}_0$ having a home configuration at the center, a data sample $\mathbf{D}_0$ of movements (shown in dash curves) generated by randomly parameterized control signals is collected. (b) The exploration starts from region $\mathcal{S}_0$. In the try-to-reach process, the robot attempts to reach the center of the adjacent region $\mathcal{S}_1$ in a finite number of trials (shown in concrete curves). (c) After collecting the best trials (shown in dash curves) of point-to-point the designated regions (shaded regions, $\mathcal{S}_1$, $\mathcal{S}_2$ and $\mathcal{S}_4$ in this case), the robot moves to and stay at the center of the adjacent region $\mathcal{S}_1$ (shown in the concrete curve), which is the new starting region. (d) The robot attempts to reach a designated region $\mathcal{S}_2$. Exploration proceeds by iteratively running the try-to-reach process in (b) to explore designated adjacent regions and the stable point positioning process in (c) to switch to new starting region.

region centers in the neighborhood. In order to obtain a data sample of end effector movements with different directions, even division method is adopted such that all regions $\mathcal{S}_l, l = 0, ..., R-1$ are square regions of equal size. The regions $\mathcal{S}_n$ with respect

to a starting region $\mathcal{S}_m$ are chosen from the regions in the neighborhood such that:

$$\begin{aligned}
&\mathcal{S}_n \in \left\{\mathcal{S}_l : d_l \leq \tilde{d}, \theta_l \leq \tilde{\theta}\right\} \\
&\vec{\boldsymbol{v}}_l : \text{Vector from the center of } \mathcal{S}_m \text{ to the center of } \mathcal{S}_l \\
&d_l = \|\vec{\boldsymbol{v}}_l\|_2, \quad \theta_l = \min\left(\angle(\vec{\boldsymbol{v}}_l, \{\vec{\boldsymbol{v}}_p, \forall p, p \neq l\})\right)
\end{aligned} \tag{5.6}$$

where d_l is the Euclidean distance between the two centers of $\mathcal{S}_m$ and $\mathcal{S}_l$. θ_l is the minimum angle between the vector $\vec{\boldsymbol{v}}_l$ and all other vector $\vec{\boldsymbol{v}}_p, p = 0, ..., R-1$. $\tilde{d}$ and $\tilde{\theta}$ constrain the traveling distance and moving direction. $\tilde{d}$ must not be smaller than the minimum distance between centers of two adjacent regions. $\tilde{\theta}$ must be smaller than $\pi/2$ in order to obtain data sample of diverse end effector movements (at least moving in orthogonal directions). For example, in a 2D task space, setting $\tilde{d} = 5$cm and $\tilde{\theta} = \frac{\pi}{2}$ means the centers of all $\mathcal{S}_n$ are 5cm away from the center of $\mathcal{S}_m$ and located in four orthogonal directions. The exploring order (i.e. the order of being starting region) of S_l is also defined. The only restriction is that two successive starting regions S_m and S_{m+1} must be connected.

It is also necessary to collect a data sample $\mathbf{D}_0$ for the first try-to-reach process at the start of exploration. The method described in section 4.2.1 is employed to generate end effector movements by applying Gaussian-like control signals $u_p(t) = A_p \exp\left(-(t - c_p)/(d_p)\right)$ (the same as (4.3)) to the pth actuator, where A_p, c_p and d_p were randomly initialized before each movement generation. These control signals produced point-to-point movements. A subset of the movements is selected according to the end effector efficiencies in orthogonal directions in the task space, defined as the ratio of distance travelled and total control input. The selection process is to ensure that the training data contained movements does not bias to some particular directions in the task space. The selected movements containing sample points of control signals $\{\boldsymbol{u}_i\}$ and the kinematics data $\{\tilde{\boldsymbol{x}}_i, \ddot{\boldsymbol{x}}_i, \boldsymbol{y}_i, \dot{\boldsymbol{y}}_i, \ddot{\boldsymbol{y}}_i\}$ are collected in a data sample $\mathbf{D}_0$. An illustrative example is depicted in Fig. 5-2a.

Table 5.1: Computing a desired trajectory of a natural reaching-like movement

---

Procedure for generating a desired trajectory for the presented exploration scheme is provide as follows.

Given a starting position $\boldsymbol{y}_m$ and a desired position $\boldsymbol{y}_n$.
Given the Fitts's Law parameters a, b, w.
Given speed profile parameters v_a, v_b.

Step 1 Compute the joint configurations $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$ for $\boldsymbol{y}_m$ and $\boldsymbol{y}_n$, respectively, such that the $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$ are the configurations farthest from the joint limits.

Step 2 Compute a joint trajectory $\{\boldsymbol{x}'_k\}_{k=1}^{\bar{N}}$ by interpolating between $\boldsymbol{x}_s$ and $\boldsymbol{x}_n$. Compute a corresponding position trajectories $\{\boldsymbol{y}'_k\}_{k=1}^{\bar{N}}$ where $\boldsymbol{y}'_1 = \boldsymbol{y}_m$ and $\boldsymbol{y}'_{\bar{N}} = \boldsymbol{y}_n$. k is the time index.

Step 3 Compute the total distant travelled $d = \sum_{k=1}^{\bar{N}-1} \|\boldsymbol{y}'_{k+1} - \boldsymbol{y}'_k\|_2$. Determine the movement time T by the Fitts's Law: $T = a + b \log_2 (1 + 2d/w)$.

Step 4 Compute a bell-shaped speed profile $\dot{d}_k = v_a \exp(-\frac{((k-1)\Delta t - T/2)^2}{2v_b^2})$ with top speed at $T/2$, and zero speed at $k = 1$ and $k = \bar{N}$, where Δt is the incremental time step and $T = (\bar{N})\Delta t$. $v_b = T/5$, v_a is optimized such that $\sum_{k=1}^{\bar{N}} \dot{d}_k = d$.

Step 5 Compute the desired position trajectory $\{\boldsymbol{y}^*_k\}_{k=1}^{\bar{N}}$ by interpolation $\{\boldsymbol{y}'_k\}_{k=1}^{\bar{N}}$ to match the speed profile $\dot{d}_k$.

Step 6 Compute the corresponding desired velocity trajectory $\{\dot{\boldsymbol{y}}^*_k\}_{k=1}^{\bar{N}}$ and $\{\ddot{\boldsymbol{y}}^*_k\}_{k=1}^{\bar{N}}$.

5.2.4 The try-to-reach process

Procedure

The detailed procedure of the try-to-reach process is described as follows. When the robot is resting at the center $\boldsymbol{y}_m$ of a starting region $\mathcal{S}_m$, it attempts to reach the center $\boldsymbol{y}_n$ of a region $\mathcal{S}_n$ in the neighborhood of $\mathcal{S}_m$ by following a desired trajectory $\mathbf{Y}^{mn}$ connecting the two centers for q_{max} trials, using an inverse mapping ${}^{\boldsymbol{a}}\zeta_{task}$ that computes low-dimensional synergy activations. The generation of the $\mathbf{Y}^{mn}$ is described in Table 5.1. Precisely, at the beginning of the process, a local data sample $\bar{\mathbf{D}}$ is selected from a subset of the library $\mathbf{D}^{explored}$ which stores collected sample points in previous try-to-reach process, such that the selected sample points locate in the neighborhood of $\boldsymbol{y}_m$:

$$\bar{\mathbf{D}} = \left\{ \mathbf{q}_i : \|\boldsymbol{y}_i - \boldsymbol{y}_m\|_2 \leq d^{local}, \boldsymbol{y}_i \in \mathbf{q}_i, \mathbf{q}_i \in \mathbf{D}^{explored} \right\} \tag{5.7}$$

where d^{local} is a predefined local distance defining the radius of the neighborhood of the center $\boldsymbol{y}_m$. $\mathbf{D}_0$ is used instead in the first try-to-reach process where $\mathbf{D}^{explored}$ is empty. A set of local synergies $\mathbf{W}^{\bar{\mathbf{D}}} \in \Re^{n^u \times M}$ and its complement set $\mathbf{W}_{\perp}^{\bar{\mathbf{D}}} \in \Re^{n^u \times (n^u - M)}$, the corresponding synergy activation $\boldsymbol{a}_i \in \Re^M$ and the mean $\bar{\boldsymbol{u}}'$ are then extracted from the local data sample $\bar{\mathbf{D}}$ using the procedure described in Table 4.1 (Step 1, Step 2 and Step 3), where the forward mapping $\psi_{task}^{\bar{D}}$ is also obtained. The number of synergies M to be utilized during the process is determined such that the total variance of $\bar{\mathbf{D}}$ explained by the first M principal components is higher than a threshold percentage ρ (e.g. 90%). These synergies in $\mathbf{W}^{\bar{\mathbf{D}}}$ are fixed in all the q_{max} trials. The task space accelerations $\ddot{\boldsymbol{y}}_i$ in $\bar{\mathbf{D}}$ are re-estimated using the sample points of the extracted synergy activation $\boldsymbol{a}_i$ and the corresponding robot state $\tilde{\boldsymbol{x}}_i$ as input to the forward mapping $\psi_{task}^{\bar{D}}$:

$$\begin{aligned} \boldsymbol{u}_i &= \mathbf{W}^{\bar{\mathbf{D}}} \boldsymbol{a}_i + \bar{\boldsymbol{u}}' \\ \ddot{\boldsymbol{y}}_i' &= \psi_{task}^{\bar{D}}(\tilde{\boldsymbol{x}}_i, \boldsymbol{u}_i) \\ \ddot{\boldsymbol{y}}_i &= \ddot{\boldsymbol{y}}_i', \quad \forall \ddot{\boldsymbol{y}}_i \in \bar{\mathbf{D}}. \end{aligned} \tag{5.8}$$

Before each trial, the inverse mapping ${}^{\boldsymbol{a}}\zeta^{\bar{\mathbf{D}}}_{task}$ is updated using $\bar{\mathbf{D}}$. During each trial, the inverse mapping ${}^{\boldsymbol{a}}\zeta^{\bar{\mathbf{D}}}_{task}$ computes synergy activation to follow the trajectory $\mathbf{Y}^{mn} = \{\boldsymbol{y}^*_k, \dot{\boldsymbol{y}}^*_k, \ddot{\boldsymbol{y}}^*_k, \boldsymbol{x}^*_k, \dot{\boldsymbol{x}}^*_k, \ddot{\boldsymbol{x}}^*_k\}_{k=1}^{\bar{N}}$ by achieving instantaneous desired task space acceleration $\ddot{\boldsymbol{y}}^*(t)$ using robot state at time step k:

$$\boldsymbol{a}_k = {}^{\boldsymbol{a}}\zeta^{\bar{\mathbf{D}}}_{task}(\tilde{\boldsymbol{x}}_k, \ddot{\boldsymbol{y}}^*_k). \tag{5.9}$$

The control input to the robot at time step k is transformed back to original control space $\mathcal{U}$ utilizing synergies $\mathbf{W}^{\bar{\mathbf{D}}}$ and $\mathbf{W}^{\bar{\mathbf{D}}}_{\perp}$:

$$\boldsymbol{u}_k = \mathbf{W}^{\bar{\mathbf{D}}}\boldsymbol{a}_k + \bar{\boldsymbol{u}}' + \mathbf{W}^{\bar{\mathbf{D}}}_{\perp}\boldsymbol{\varsigma} \tag{5.10}$$

where $\boldsymbol{\varsigma}$ is a Gaussian random vector with compatible size drawn before each trial, and is constant during each trial. The sample points $\mathbf{D}^q = \{\mathbf{q}_k\}_{k=1}^{\bar{N}}$ of the jth trial are appended to $\bar{\mathbf{D}}$:

$$\bar{\mathbf{D}} = \bar{\mathbf{D}} \cup \mathbf{D}^q \tag{5.11}$$

The robot is reset back to the center of $\mathcal{S}_m$ after each trial. The resetting of the end effector is achieved by using a feedback controller, which will be described in section 5.2.5 later. After q_{max} trials, the data sample of the best trial $\mathbf{D}^{q^*}$ having the smallest tracking position error $e_q = \sum_k(\|\boldsymbol{y}_k - \boldsymbol{y}^*\|_2)$ is selected and appended to the set $\mathbf{D}^{explored}$.

$$\mathbf{D}^{explored} = \mathbf{D}^{explored} \cup \mathbf{D}^{q*}, \quad q^* = \arg\min_q(e_q). \tag{5.12}$$

Fig. 5-2b and Fig. 5-2d illustrate the try-to-reach process.

Elaboration

During a try-to-reach process, the control input (5.10) to the robot is computed using the inverse mapping ${}^{\boldsymbol{a}}\zeta^{\bar{\mathbf{D}}}_{task}$ with exploratory noise to move the end effector closer to the center $\boldsymbol{y}_n$ of $\mathcal{S}_n$. The first two terms $\mathbf{W}^{\bar{\mathbf{D}}}\boldsymbol{a}_k + \bar{\boldsymbol{u}}'$ in (5.10) transform the low-dimensional synergy activation back to the original control space $\mathcal{U}$ using (5.2). It has been proven in chapter 4 that low-dimensional inverse mapping ${}^{\boldsymbol{a}}\zeta_{task}$ can be used as

a feedforward control to track a task space trajectory $\mathbf{Y}' = \{\boldsymbol{y}_k^*, \dot{\boldsymbol{y}}_k^*, \ddot{\boldsymbol{y}}_k^*\}_{k=1}^{\bar{N}}$ with the aid of synergies, if the inverse mapping ${}^{\boldsymbol{a}}\zeta_{task}$ is accurate enough. In a try-to-reach process, more sample points are added to the local data sample $\bar{\mathbf{D}}$ at the end of each trials, resulting in improving the accuracy of the inverse mapping ${}^{\boldsymbol{a}}\zeta_{task}^{\bar{\mathbf{D}}}$ and thus can bring the end effector close to the center $\boldsymbol{y}_n$ of $\mathcal{S}_n$ by tracking the trajectory $\mathbf{Y}^{mn}$ with desired end effector position $\boldsymbol{y}^* = \boldsymbol{y}_n$. In order to produce data sample of "natural" human-like movements, the Fitts's law is adopted in generating the desired trajectory $\mathbf{Y}^{mn}$ such that the velocity profile of the end effector is determined from a human-like traveling time.

The last term $\mathbf{W}_{\perp}^{\bar{\mathbf{D}}}\boldsymbol{\varsigma}$ in (5.10) adds exploratory noise such that the control input lies not only in the subspace by the local synergies $\mathbf{W}^{\bar{\mathbf{D}}}$, but in the original control space $\mathcal{U}$. The addition of the exploratory noise is theoretically essential to the success of the exploration. Because the sample points of the control signals are used as sample points (5.11) and (5.12) for synergy extraction in another try-to-reach process that moving to different region center, say $\boldsymbol{y}_{n'}$, the control input to reach $\boldsymbol{y}_{n'}$ by (5.10) will be in the same subspace spanned by $\mathbf{W}^{\bar{\mathbf{D}}}$ if there is no exploratory noise. That is the local synergies $\mathbf{W}^{\bar{\mathbf{D}}}$ will be fixed for all try-to-reach processes in the whole task space $\mathcal{S}$. This is undesirable because different local synergies, which are groups of co-activation of actuators, should be utilized for moving to targets in different/distal regions in $\mathcal{S}$.

The re-estimation step in (5.8) is a necessary step to ensure that the sample points $\ddot{\boldsymbol{y}}_i$ are the task space accelerations of applying $\boldsymbol{a}_i$ at $\tilde{\boldsymbol{x}}_i$. In chapter 4, this consistency is achieved by regenerating corresponding end effector movements using extracted sequences of $\boldsymbol{a}_i$. In this chapter, this step is carried out by prediction using the forward mapping $\psi_{task}^{\bar{\mathbf{D}}}$, because collecting new set of data sample by regeneration is costly.

It is noteworthy that because the objective of the exploration scheme is to generate sample points of end effector movements with different moving directions and spreading over the whole space $\mathcal{S}$, it is not necessary to accurately track the desired trajectory $\mathbf{Y}^{mn}$. If the noise level is not too high, the robot can still move the end

effector close to the center $\boldsymbol{y}_n$ of $\mathcal{S}_n$.

With the aid of synergies, computation spent in the estimation of the inverse mapping ${}^{\boldsymbol{a}}\zeta_{task}$ can be reduced. After synergy extraction, the synergy activations $\boldsymbol{a}_i$ become independent in each dimension after performing PCA on the sample points of the control signals $\boldsymbol{u}_i$. The inverse mapping ${}^{\boldsymbol{a}}\zeta_{task}$ can thus be estimated by multiple univariate regressions. Computation can be reduced by running fewer univariate regressions.

5.2.5 The stable point positioning process

In the proposed exploration scheme, the robot needs to reset the end effector to a stable point at either the center $\boldsymbol{y}_m$ of a current starting region $\mathcal{S}_m$ at the end of each trial, or to the center $\boldsymbol{y}_{m'}$ of a new starting region $\mathcal{S}_{m'}$. Since the robot has to start from rest during the try-to-reach process, this resetting procedure is named as the stable point positioning process in this chapter. The positioning process can be achieved using a feedback controller to manipulate the end effector to follow a desired trajectory connecting from the current end effector location $\boldsymbol{y}_{current}$ with robot state $\tilde{\boldsymbol{x}}_{current}$ to the center $\boldsymbol{y}_m$ or $\boldsymbol{y}_{m'}$ with zero end effector velocity for a certain duration of the last time period. The joint configuration at the start of each trial in the try-to-reach process is also influential to the goal-directed exploration. It has been discussed in [137] that hitting joint limits during exploration can lead to failure of further exploration as the inverse mapping cannot be improved. Therefore, the controller should also capable of regulating the joints keeping away from the joint limits while manipulating the end effector in task space.

A feedback task space controller $\Pi^{\hat{\mathbf{D}}}$ is employed to position the end effector to the region center $\boldsymbol{y}_m$ or $\boldsymbol{y}_{m'}$ and to keep the joints away from joint limits. The controller $\boldsymbol{u}_k = \Pi^{\hat{\mathbf{D}}}\left(\tilde{\boldsymbol{x}}_k, \ddot{\boldsymbol{y}}_k, \boldsymbol{e}_k, \dot{\boldsymbol{e}}_k, \boldsymbol{\varepsilon}(\boldsymbol{x}_k)\right)$ is developed in chapter 4, in which the control input to the robot is computed by the low dimensional inverse mapping ${}^{\boldsymbol{a}}\zeta_{task}^{\hat{\mathbf{D}}}$ obtained from a data sample $\hat{\mathbf{D}}$ with the aid of synergies $\mathbf{W}^{\hat{\mathbf{D}}}$. It is capable of manipulating the robot end effector to follow a desired task space trajectory $\{\boldsymbol{y}_k^*, \dot{\boldsymbol{y}}_k^*, \ddot{\boldsymbol{y}}_k^*\}_{k=1}^{\bar{N}}$. Simultaneously, joint stabilization and a secondary task goal defined in a function

$\varepsilon(\boldsymbol{x}_k)$ to keep the joints away from joint limits can be achieved without affecting the tracking performance. A human-like trajectory is used as in the try-to-reach process as mentioned in section 5.2.4, such that $\boldsymbol{y}_1^* = \boldsymbol{y}_{current}$ and $\boldsymbol{y}_{\bar{N}}^* = \boldsymbol{y}_m$ (or $\boldsymbol{y}_m$) in the desired trajectory $\mathbf{Y}^* = \{\boldsymbol{y}_k^*, \dot{\boldsymbol{y}}_k^*, \ddot{\boldsymbol{y}}_k^*\}_{k=1}^{\bar{N}}$. The desired trajectory is determined using the method described in Table 5.1, such that the corresponding joint angles $\{\boldsymbol{x}_k^*\}_{k=1}^{\bar{N}}$ are farthest away from the joint limits. $\mathbf{Y}^*$ is appended by repeated values of $\boldsymbol{x}_{\bar{N}}^*$ and $\boldsymbol{y}_{\bar{N}}^*$ with zero velocities in order to stay at $\boldsymbol{y}_m$ or $\boldsymbol{y}_{m'}$. The control input $\boldsymbol{u}_k$ at time t is given by:

$$\begin{aligned} \boldsymbol{a}_k &= {}^{\boldsymbol{a}}\zeta_{task}^{\hat{\mathbf{D}}}\left(\tilde{\boldsymbol{x}}_k, \ddot{\boldsymbol{y}}_k\right) + \Omega_{fb}\left(\tilde{\boldsymbol{x}}_k, \boldsymbol{e}_k, \dot{\boldsymbol{e}}_k\right) + \Omega_{null}\left(\tilde{\boldsymbol{x}}_k, \varepsilon(\boldsymbol{x}_k)\right) \\ \boldsymbol{u}_k &= \mathbf{W}^{\hat{\mathbf{D}}}\boldsymbol{a}_k + \bar{\boldsymbol{u}}^{\hat{\mathbf{D}}} \end{aligned} \tag{5.13}$$

where $\boldsymbol{e}_k = \boldsymbol{y}_k - \boldsymbol{y}_k^*$ and $\dot{\boldsymbol{e}}_k = \dot{\boldsymbol{y}}_k - \dot{\boldsymbol{y}}_k^*$ are the tracking position error and the velocity error, respectively. $\bar{\boldsymbol{u}}^{\hat{\mathbf{D}}}$ is the mean of control signal in the data sample $\hat{\mathbf{D}}$. The first term ${}^{\boldsymbol{a}}\zeta_{task}^{\hat{\mathbf{D}}}\left(\tilde{\boldsymbol{x}}_k, \ddot{\boldsymbol{y}}_k\right)$ is the feedforward control term responsible for achieving tracking the desired trajectory. The second term $\Omega_{fb}\left(\tilde{\boldsymbol{x}}_k, \boldsymbol{e}_k, \dot{\boldsymbol{e}}_k\right)$ is the feedback control term responsible for reducing the tracking error in the task space. The third term $\Omega_{null}\left(\tilde{\boldsymbol{x}}_k, \varepsilon(\boldsymbol{x}_k)\right)$ is the null space control term is responsible for joint stabilization and achieving the secondary goal defined in $\varepsilon(\boldsymbol{x}_k)$:

$$\varepsilon\left(\boldsymbol{x}_k\right) = \boldsymbol{x}_k - \boldsymbol{x}_k^*. \tag{5.14}$$

such that the joint angles $\boldsymbol{x}$ is "pulled" to the desired joint angle $\boldsymbol{x}_k^*$ away from the joint limits. Ω_{null} is formulated with the aid of the inverse dynamics mapping ${}^{\boldsymbol{a}}\zeta_{joint}$ in joint space. The synergy activation $\boldsymbol{a}_k$ is transformed back to the control space $\mathcal{U}$

using (5.2). The detailed formulations of the controller in (5.13) are:

$$
{}^{\boldsymbol{a}}\zeta_{task}\left(\tilde{\boldsymbol{x}}_k, \ddot{\boldsymbol{y}}_k\right) = \tilde{\hat{\beta}}^{\dagger}\left(\tilde{\boldsymbol{x}}_k\right)\left[\ddot{\boldsymbol{y}}_k^{*} - \tilde{\hat{\alpha}}(\tilde{\boldsymbol{x}}_k)\right] \tag{5.15}
$$

$$
\Omega_{fb}(\tilde{\boldsymbol{x}}_k, \boldsymbol{e}_k, \dot{\boldsymbol{e}}_k) = \tilde{\hat{\beta}}^{\dagger}(\tilde{\boldsymbol{x}}_k)\left(\boldsymbol{k}_s \circ \mathrm{sat}(\boldsymbol{s}_k) + \lambda \dot{\boldsymbol{e}}_k\right) \tag{5.16}
$$

$$
\Omega_{null}\left(\tilde{\boldsymbol{x}}_k, \hat{\varepsilon}(\boldsymbol{x}_k)\right) = \left[\mathbf{I} - \tilde{\hat{\beta}}^{\dagger}(\tilde{\boldsymbol{x}}_k)\tilde{\hat{\beta}}(\tilde{\boldsymbol{x}}_k)\right]\boldsymbol{a}^{null}\left(\tilde{\boldsymbol{x}}_k, \hat{\varepsilon}(\boldsymbol{x}_k)\right) \tag{5.17}
$$

$$
\boldsymbol{a}^{null}\left(\tilde{\boldsymbol{x}}_k, \hat{\varepsilon}(\boldsymbol{x}_k)\right) = \tilde{g}^{\dagger}(\tilde{\boldsymbol{x}}_k)\left[-\mathbf{K}_{v0}\dot{\boldsymbol{x}}_k - \mathbf{K}_{p0}\hat{\varepsilon}(\boldsymbol{x}_k)\right] \tag{5.18}
$$

where $\boldsymbol{s}_k = \dot{\boldsymbol{e}}_k - \lambda \boldsymbol{e}(t)$. $\hat{\alpha}\left(\tilde{\boldsymbol{x}}_k\right)$, $\hat{\beta}\left(\tilde{\boldsymbol{x}}_k\right)$ and $\hat{g}\left(\tilde{\boldsymbol{x}}_k\right)$ the functions in (5.4) and (5.5). $\tilde{\hat{\beta}}\left(\tilde{\boldsymbol{x}}_k\right)$ is the estimate of $\hat{\beta}(\tilde{\boldsymbol{x}}_k)$, $\tilde{\hat{\beta}}^{\dagger}\left(\tilde{\boldsymbol{x}}_k\right)$ and $\tilde{\hat{g}}^{\dagger}\left(\tilde{\boldsymbol{x}}_k\right)$ are the estimates of the Moore-Penrose pseudo-inverse $\hat{\beta}^{\dagger}\left(\tilde{\boldsymbol{x}}_k\right)$ and $\hat{g}^{\dagger}\left(\tilde{\boldsymbol{x}}_k\right)$, respectively. $\hat{g}\left(\tilde{\boldsymbol{x}}_k\right)$ is obtained by substituting (5.2) into (5.1). $\tilde{\hat{g}}^{\dagger}\left(\tilde{\boldsymbol{x}}_k\right)$ is obtained by estimating the inverse dynamics mapping ${}^{\boldsymbol{a}}\zeta_{joint}$ in joint space. $\mathbf{I}$ is an identity matrix, $\boldsymbol{k_s}$ is the control gain, $\mathrm{sat}(\cdot, \boldsymbol{\Psi})$ is an entry-wise saturation function, in which the jth entry equals -1, 1 if $\frac{s_j}{\Psi_j} < 1$, $\frac{s_j}{\Psi_j} > 1$, respectively and s_j otherwise. $\boldsymbol{\Psi} \in \Re^{n^y}$ is a vector of positive entries. $\circ$ denotes the Hadamard product.

The data sample $\hat{\mathbf{D}}$ defines the regions where the end effector is controllable by the controller $\Pi^{\hat{\mathbf{D}}}$. $\hat{\mathbf{D}}$ can be selected such that $\hat{\mathbf{D}}$ is as a subset of the whole data sample $\mathbf{D}^{explored}$ in the neighborhood of the current end effector location $\boldsymbol{y}_{current}$:

$$
\hat{\mathbf{D}} = \left\{\mathbf{q}_i : \|\boldsymbol{y}_i - \boldsymbol{y}_{current}\|_2 \leq \hat{d}^{local}, \boldsymbol{y}_i \in \mathbf{q}_i, \mathbf{q}_i \in \mathbf{D}^{explored}\right\} \tag{5.19}
$$

where $\hat{d}^{local}$ is a predefined local distance defining the radius of the neighborhood of the center $\boldsymbol{y}_{current}$. At the end of each trail in the try-to-reach process, if the current end effector location $\boldsymbol{y}_{current}$ lies far away from the explored regions where the sample points in $\hat{\mathbf{D}}$ are collected, the controller may not be able to manipulate the end effector back to the center $\boldsymbol{y}_m$ or $\boldsymbol{y}_{m'}$. If the end effector cannot be positioned back to the center $\boldsymbol{y}_m$ or $\boldsymbol{y}_{m'}$ using the controller $\Pi^{\hat{\mathbf{D}}}$ where the local data $\hat{\mathbf{D}}$ is obtained by (5.19), the end effector can be first reset to the home configuration $\boldsymbol{x}_0$ at region $\mathcal{S}_0$, followed by positioned to the center $\boldsymbol{y}_m$ or $\boldsymbol{y}_{m'}$ using the controller $\Pi^{\hat{\mathbf{D}}'}$ obtained from another local data sample $\hat{\mathbf{D}}'$, which is a collection of sample points in $\mathbf{D}^{explored}$ that close to

the trajectory $\mathbf{Y}^*$:

$$\begin{aligned}\hat{\mathbf{D}}' &= \left\{\mathbf{q}_i : d_i^{\mathbf{Y}^*} \leq \hat{d}^{local}, \boldsymbol{y}_i \in \mathbf{q}_i, \boldsymbol{y}_k^* \in \mathbf{Y}^*, \mathbf{q}_i \in \mathbf{D}^{explored}\right\} \\ d_i^{\mathbf{Y}^*} &= \min_k(\{\|\boldsymbol{y}_i - \boldsymbol{y}_k^*\|_2\}_{k=1}^{\bar{N}})\end{aligned} \tag{5.20}$$

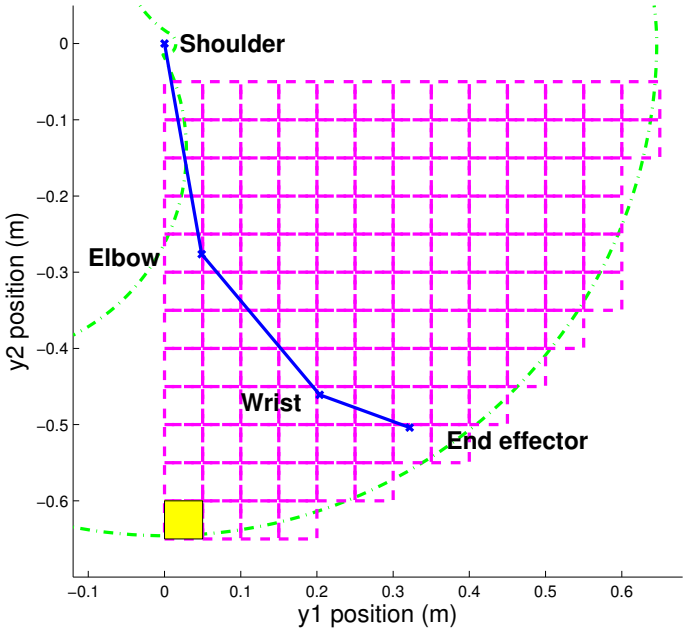
where $d_i^{\mathbf{Y}^*}$ is the minimum distance of between a sample point $\boldsymbol{y}_i$ and the trajectory $\mathbf{Y}^*$. The controller $\Pi^{\hat{\mathbf{D}}'}$ can position the end effector since the home configuration $\boldsymbol{x}_0$ and the center $\boldsymbol{y}_m$ or $\boldsymbol{y}_{m'}$ are inside the explored regions where the data sample $\mathbf{D}^{explored}$ were collected. Fig. 5-2c gives an illustrative example of moving to a stable point.

5.3 Experiments

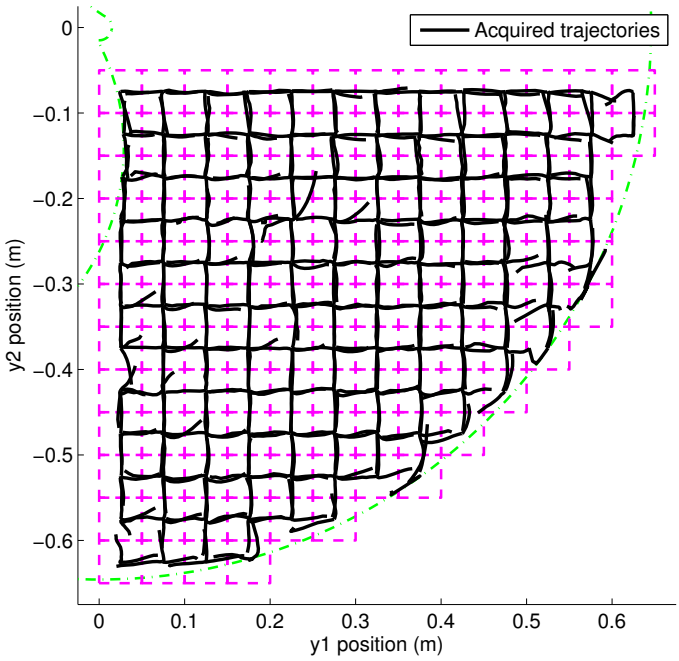
This section presents simulation results of an exploration task and two control tasks on a human-like robotic arm on a 2D task space. The robotic arm had 3 joints and actuated by 10 muscles. The muscles produce torque on the joints that satisfies the linear relation (4.2). Fig. 5-3a depicts the skeleton of the robotic arm on the task space which were divided into 122, 5cm $\times$ 5cm square regions. The exploration task was conducted to verify if the proposed scheme utilizing synergies is feasible for exploration of all the 122 regions. The control tasks were carried out to examine if synergies can be extracted from the data sample collected by the exploration scheme for controlling the robot. In both of the tasks, measurements of the joint angles $\boldsymbol{x}$, angular velocities $\dot{\boldsymbol{x}}$, angular accelerations $\ddot{\boldsymbol{x}}$, end effector position $\boldsymbol{y}$, velocities $\dot{\boldsymbol{y}}$ and accelerations $\ddot{\boldsymbol{y}}$ were corrupted by Gaussian noise with standard deviation of 0.2°, 2°s^{-1}, 20°s^{-2}, 0.002m, 0.02ms^{-1} and 0.2ms^{-2}, respectively. In the estimations, a Gaussian kernel: $k(\tilde{\boldsymbol{x}}_p, \tilde{\boldsymbol{x}}_q) = \exp\left(-\frac{1}{2}(\tilde{\boldsymbol{x}}_p - \tilde{\boldsymbol{x}}_q)^T\mathbf{L}(\tilde{\boldsymbol{x}}_p - \tilde{\boldsymbol{x}}_q)\right)$ was employed, where $\mathbf{L}$ was a diagonal matrix. The mth entry at the diagonal was $L_m = \sigma^{-1}(\tilde{x}_m^{max} - \tilde{x}_m^{min})^{-2}$ where σ was a scalar defining the kernel width. Muscle synergies were extracted by performing PCA using the MATLAB function *pca*. All simulations were carried out on a computer with 2.8 Ghz Intel Core i7, 16GB 1600 MHz DDR3.

5.3.1 Exploration task

The exploration started from the lowest left region $\mathcal{S}_0$ in the grid depicted in Fig. 5-3a. In the initialization process, $\tilde{d} = 5$cm and $\tilde{\theta} = \pi/2$ was set such that in a starting region $\mathcal{S}_m$, the robot end effector was controlled to reach the centers of neighbor regions 5cm away from the current region center in the y_1 and y_2 directions in the trying-to-reach process. The exploration order of the starting region $\mathcal{S}_m$ was chosen from the left to right, and then from lower to upper. The top right region was the last region being explored. In collecting the data sample $\mathbf{D}_0$ for the first try-to-reach process, 100 movements of traveling distance around 10cm (able to cover the regions connected to $\mathcal{S}_0$) were generated by the randomly initialized parameterized control signals (4.3) as described in section 5.2.3. Each movement had the same number of sample points. 8 of the 100 movements that having the largest traveling distance in the $\pm y_1$ and $\pm y_2$ direction were collected in the $\mathbf{D}_0$. $\mathbf{D}_0$ had 120 sample points. In each try-to-reach process, sample points that located within a radius of $d^{local} = 5$cm from the center $\boldsymbol{y}_m$ of the starting region $\mathcal{S}_m$ were collected in the local data sample $\bar{\mathbf{D}}$ for synergy extraction and estimating the inverse mapping ${}^{\boldsymbol{a}}\zeta^{\bar{\mathbf{D}}}_{task}$. M synergies were extracted from $\bar{\mathbf{D}}$ such that the synergies in $\mathbf{W}^{\bar{\mathbf{D}}} \in \Re^{10\times M}$ explained at least $\rho = 90\%$ of the total variance in $\bar{\mathbf{D}}$. $q_{max} = 30$ trials were attempted to reach the center $\boldsymbol{y}_n$ of the designated region $\mathcal{S}_n$ in the neighborhood of $\mathcal{S}_m$. Before each trial, the Gaussian random vector $\boldsymbol{\varsigma}$ of 0.2 standard deviation was drawn to activate the remaining $10 - M$ synergies in $\mathbf{W}^{\bar{\mathbf{D}}}_{\perp}$. In the try-to-reach processes where $\mathcal{S}_0$ was the start region, the estimation parameters σ (kernel width) and the regularization constant γ for obtaining the inverse mapping ${}^{\boldsymbol{a}}\zeta^{\bar{\mathbf{D}}}_{task}$ were determined by a trial-and-error process such that the centers $\boldsymbol{y}_n$ of designated regions $\mathcal{S}_n$ could be approximately reached. The same parameters were then used throughout the whole exploration task. As discussed in section 5.2.5, if the end effector locates far away from the center $\boldsymbol{y}_m$ of the current region $\mathcal{S}_m$ at the end of a trial, a feedback controller $\Pi^{\hat{\mathbf{D}}}$ constituted from a local sample $\hat{\mathbf{D}}$ (see (5.19)) may not be able to position the end effector back to $\boldsymbol{y}_m$. For the ease of implementation and analysis, at the end of all trials in the simulation,



(a)



(b)

Figure 5-3: (a) A 3-links human-like robotic arm on the 2D task space. The skeleton of the robot is shown in concrete lines. The dash-dot curves are the boundaries of the reachable area. The grid is the task space in the exploration task. The robot starts exploration from the shaded lowest left region where the home configuration of the robot locates. (b) The acquired movements after exploration.

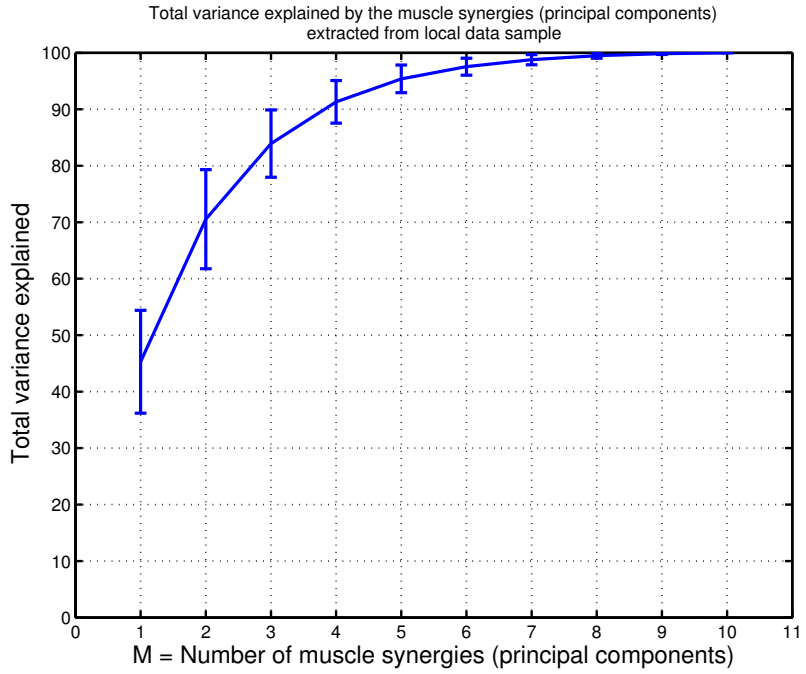


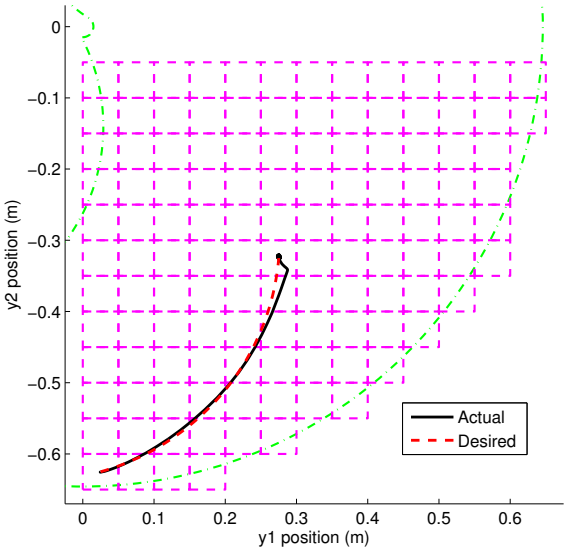
Figure 5-4: Variance explained by the principal components (muscle synergies) extracted from local data samples $\bar{\mathbf{D}}$ during exploration. The line is the mean of the total variance explained by the first M principal components and the error bars have length of two standard deviation units.

the end effector was first reset back to the home configuration $\boldsymbol{x}_0$ at $\mathcal{S}_0$, followed by positioned to a stable point at either $\boldsymbol{y}_m$ or $\boldsymbol{y}_{m'}$ using the controller $\Pi^{\hat{\mathbf{D}}'}$ constituted from a subset $\hat{\mathbf{D}}'$ having sample points close to the trajectory ($\hat{d}^{local}$ = 10cm) from the home configuration to the stable point $\boldsymbol{y}_m$ or $\boldsymbol{y}_{m'}$.

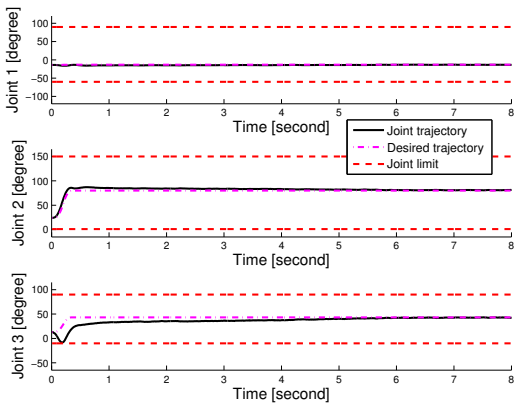
Fig. 5-3b shows the movements acquired in the data sample $\mathbf{D}^{explored}$ after exploration. These movements were collected in the data sample $\mathbf{D}^{explored}$. It can be observed that the robot successfully explored the task space. Fig. 5-4 depicts the variance explained by the synergies (principal components) extracted from the local data samples $\bar{\mathbf{D}}$. The number of synergies utilized M during exploration was determined such that the M synergies could explain 90% of the total variance. It was observed that the exploration task was conducted utilizing 4 or 5 synergies (mostly 4) to the trajectories depicted in Fig. 5-3b.

5.3.2 Control performance

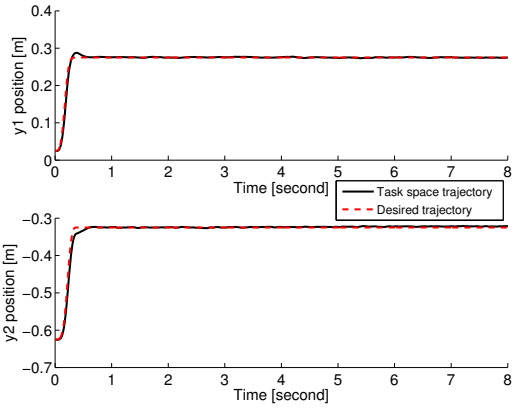
Two control tasks were carried out. In the first control task was carried out during the exploration when the robot was positioned to the centers of all the 122 regions from the home configuration $\boldsymbol{x}_0$. The local data sample $\bar{\mathbf{D}}'$ for constituting a controller $\Pi^{\hat{\mathbf{D}}'}$ was a subset of $\mathbf{D}^{explored}$ at that exploration moment, which consists of sample points close to the trajectory $\mathbf{Y}^{1,n}$ ($\hat{d}^{local}$ = 10cm) connecting from the home configuration $\boldsymbol{x}_0$ in the region $\mathcal{S}_1$ to the center of region $\mathcal{S}_n$. The second task was carried to verify if the extracted synergies and the collected data sample can be used for constituting a controller for controlling the robot in a reduced control dimensionality within the whole explored task space. The robot needed to track a trajectory of a figure of "8" in the task space depicted in Fig. 5-6a, utilizing a set of synergies extracted from the data sample $\mathbf{D}^{explored}$ collected after exploration. The secondary control goal in the null space control was to keep the joints away from the joints limit. Without given desired trajectory in the joint space, the function $\varepsilon(\boldsymbol{x}_k)$



(a) Trajectory on the task space

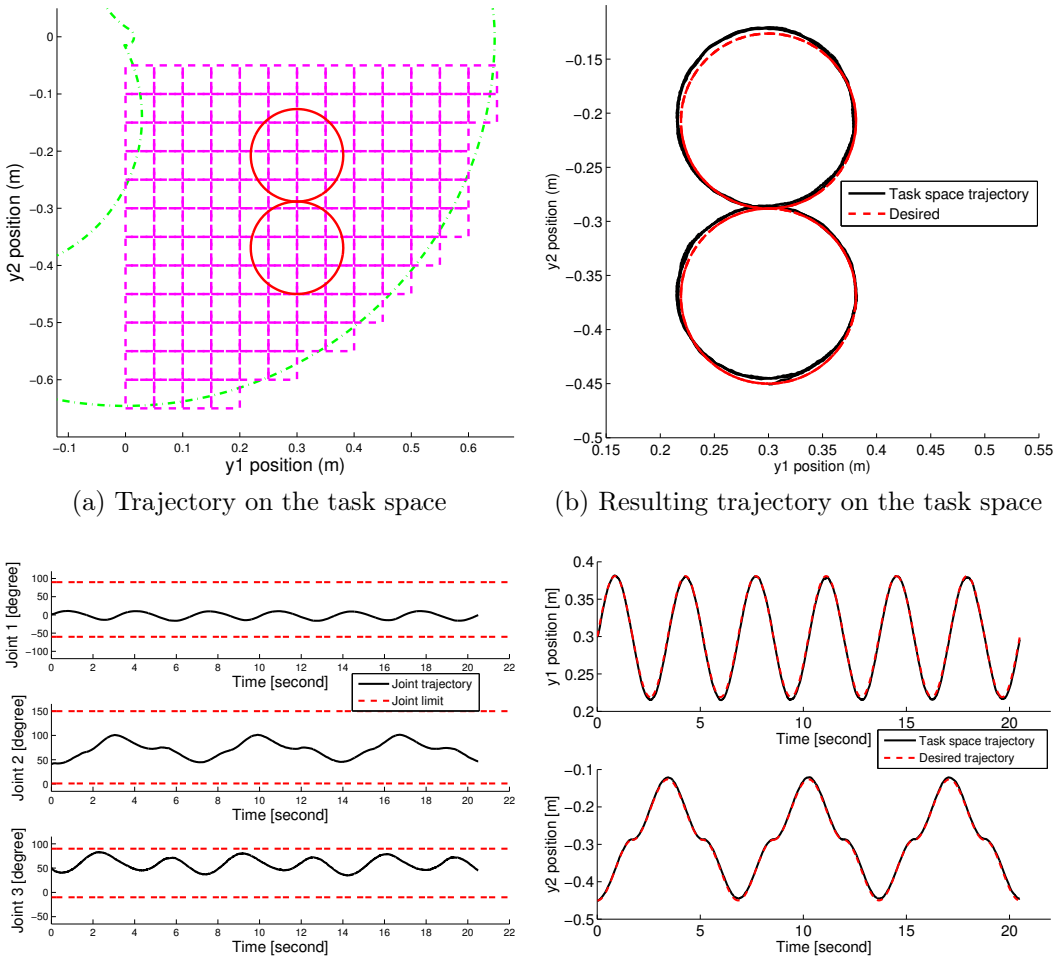


(b) Joint space history



(c) Task space history

Figure 5-5: Positioning of the end effector to a stable point in the task space during exploration. The end effector could stay at the target position with all joints being regulated to the desired joint angles by the feedback controller associated with null space control term.



(a) Trajectory on the task space

(b) Resulting trajectory on the task space

(c) Joint space history

(d) Task space history

Figure 5-6: Controlling the robot utilizing a set of global muscle synergies extracted from the data sample $\mathbf{D}^{explored}$ after exploration. 5 synergies were utilized in the tracking. It was observed that the desired trajectory could be followed and the joints were kept within the joint limits by the null space control.

in (5.14)) was replaced by the same function defined in (4.26):

$$\hat{\varepsilon}_{lk} = \begin{cases} x_{lk} - (x_l^{lb} + \delta), & x_l^{lb} \leq x_{lk} \leq x_l^{lb} + \delta \\ x_{lk} - (x_l^{ub} - \delta), & x_l^{ub} - \delta \leq x_{lk} \leq x_l^{ub} \\ 0 & \text{otherwise} \end{cases} \tag{5.21}$$

where $\hat{\varepsilon}_{lk}$ indicates the lth entry of $\boldsymbol{x}_k$. δ is a positive scalar.

Fig. 5-5 depicts a resulting trajectory of moving to a region centers. It was observed that the end effector could be accurately positioned to the target center, and the joints could also be regulated to desired joint angles. In all 122 cases of moving to the region centers, the maximum position error in the task space and in the joint space in the last 5 seconds were 0.0122m and 9.523degree, respectively. These results confirmed that the local data sample $\hat{\mathbf{D}}'_m$ and the extracted synergies $\mathbf{W}^{\hat{\mathbf{D}}'}$ can be used to constitute a low-dimensional controller to position the end effector to a stable point at the center of region $\mathcal{S}_m$ during exploration. Fig. 5-6 depicts the results of tracking the figure of "8" trajectory. The tracking was achieved by 5 synergies that explained 92% of the total variance in $\mathbf{D}^{explored}$. It was observed that the trajectory could be followed without collision to the joint limits.

5.4 Discussion

5.4.1 On the choice of the number of synergies for exploration

In section 5.3.1, the number of synergies used for the exploration task is preset before the act of control so that the selected synergies can explain a certain threshold percentage of the total data variance. In the experiment, a guess of 90% was set for the threshold.

As discussed in section 3.5.3, to determine the best number of synergies for a case, a performance index that measures the control performance of a synergies-based controller is necessary. Note that the total data variance explained depicted in Fig. 5-4 is an inappropriate measure for determining the number of synergies to carry out the

exploration since it does not reflect any control performance of the synergies-based controller. In the future work, the exploration task will be conducted by using a fixed number of synergies. The best (minimum) number of synergies may then be determined, by comparing the control performance of exploration using different number of synergies, according to a certain performance index such as tracking error. Further investigation of the best number of synergies compromising the control performance and the control dimensionality (complexity) for the exploration task will be carried out as one of the future works.

5.4.2 Time-invariant synergies vs time-varying synergies

One of the main results in this chapter was that exploration could be achieved in reduced dimensionality utilizing synergies. In [154], goal-directed exploration was carried out with the aid of time-varying synergies, which specify sequences of control signals of all actuators. There were 24 control variables to control a simulated 12 degree-of-freedom quadruped robot thus the control dimensionality was not reduced. In contrast, time-invariant synergies are utilized in the proposed method, which always has dimensionality not larger than the original one. In the simulation, 4 or 5 synergies were enough for exploration of the robot actuated by 10 muscles.

In robotic applications, feedforward controllers have been used to coordinate time-varying synergies that represent time sequences of muscle activations, before the start of a movement [17, 20, 142]. In contrast, feedback controllers have been used to coordinate time-invariant synergies by computing the synergy activations at each time step as demonstrated in chapter 4. Feedback controllers are more desirable for general manipulation and better control accuracy, while feedforward controllers are simpler and more suitable for specific tasks. The use of time-invariant synergies allows a straightforward implementation of a feedback controller with a null space control term. The null space control term regulated the joints to keep close to desired joint configurations, as a secondary control objective during exploration.

5.5 Summary

A data acquisition method is proposed that allows musculoskeletal robots to obtain muscle synergies by itself without prior knowledge about the complicated body structure. The concept of muscle synergies is a control strategy inspired by the study of human motor control. In most engineering applications, a data sample is usually given for muscle synergies extraction. The proposed method adopts a motor skill learning strategy inspired from infants called goal-directed exploration, in which the robot collects data sample of point-to-point movements spreading over the task space. During exploration, muscle synergies participate in constituting controller from local data samples to achieve the point-to-point movements. The controller computes control inputs from inverse dynamics mapping in reduced dimensionality with the aid of muscle synergies. A feedback controller is employed to support the progression of exploration such that the robot end effector can be positioned at a stable point in the task space that is a starting point of a point-to-point movement trail. As the starting joint configuration is crucial for successful exploration, a null space control term is employed to keep the joints away from the joint limits. The present method was evaluated in a simulated human-like robotic arm. The human-like robotic arm could successfully explore a 2D task space. Moreover, it was shown that the data sample collected after exploration could be used as training source to extracted a set of synergies associated with a low-dimensional controller for manipulating the robot end effector within explored task space.

Chapter 6

Conclusion

6.1 Conclusion

Musculoskeletal robots have a redundant number of joints and muscles that can perform a diversity of tasks. However, the complex body structure leads to difficulties in control. One control difficulty arises from the many degree-of-freedom body structure. Dimensionality reduction is crucial to develop methods to effectively control the robots. Muscle synergies have been considered as one of the solutions to the degree-of-freedom problem of human body control in biological studies. This research studies control methods using muscle synergies for musculoskeletal robots. On the whole, this research contributes to the investigation of the feasibility of control methods utilizing muscle synergies for a musculoskeletal robot. This research would be the first step to the realization of robots that can work in daily life.

Chapter 3 studies the feasibility of utilizing muscle synergies in controlling a musculoskeletal robot. In simulation experiments, muscle synergies extracted from different data samples of control signals that were optimized according to specific criteria were used to generate omnidirectional reaching movements of the end effector of a human-like robotic arm actuated by 10 muscles. It was found that 1) the achieving-goal synergies, which were extracted from data sample of optimized movements that had minimum distances from targets at the final time step and minimum total control effort spent, and 2) the energy-efficient synergies, which were extracted from

data sample of movements optimized with respect to a fitness function defined as the ratio between kinetic energy and the movement and control, could be utilized to control the robot in reduced control dimensionality. These results not only imply muscle synergies can be utilized to control musculoskeletal robots in reduced control dimensionality, but also demonstrate that muscle synergies can be extracted from a data sample of optimized control signals with respect to energy efficiency, regardless of whether task goals are specified in the optimization.

Chapter 4 investigates the problem of how to extract muscle synergies given data sample without statistical regularities and without given dynamics model of a musculoskeletal robot. The problem of extracting muscle synergies from a data sample of movements that are generated by randomly parameterized control signals is considered. Because data sample with certain statistical regularities cannot be always provided in reality, but it is easier to generate such random movements. In order to tackle modeling uncertainties, a kernel-based system identification method is developed to directly estimate the robot's forward and inverse dynamics models from the data sample. In the presented method, muscle synergies are extracted from a data sample of the control signals that are estimated using the kernel-based system identification method. The estimated control signals are optimal with respect to minimum control effort (minimum Euclidean norm of the control signals). A data-driven task space tracking controller, which is capable of manipulating the robot end effector to follow a desired task space trajectory and achieving secondary control goals, is derived using nonlinear sliding mode control technique. In simulation experiments on a human-like robotic arm actuated by 10 muscles, it was shown that muscle synergies could be extracted using the proposed method but could not be extracted from the randomly parameterized control signals, and the robot could be controlled utilizing 5 muscle synergies to follow a figure of "8" trajectory on a 2D task space. Moreover, it was shown that muscle synergies could be further refined by extracting from controlled movements data. The proposed method allows the extraction of muscle synergies from a given data sample of movements where the muscle activations are statistically independent in each dimension.

Chapter 5 investigates how to extract muscle synergies without a given data sample and prior knowledge of the robot dynamics model. Although chapter 4 has developed a technique for extracting muscle synergies from data sample without statistical regularities, it requires the robot can start from anywhere to collect sample points spreading over the task space, which may not be always satisfied. A goal-directed exploration scheme is proposed such that a musculoskeletal robot can collect appropriate data sample for extraction of muscle synergies. In the proposed scheme, the robot explores the task space by reaching designated targets spreading over the task space successively, in which the robot uses inverse dynamics and muscle synergies obtained from local data to control the end effector in a reduced control dimensionality. In a simulation evaluation, the proposed scheme allowed a human-like robotic arm with 10 muscles to collect data sample by exploring a 2D task space. It was also demonstrated that muscle synergies could be extracted from the collected data sample to establish a task space tracking controller for manipulation of the robot end effector in reduced control dimensionality. The proposed goal-directed exploration scheme enables a robot to obtain muscle synergies by itself, which is a step forward to the development of autonomous musculoskeletal robots.

6.2 Limitations

Static environment

The proposed methods in this research may not be applicable to extract synergies for control in dynamic environment. In all chapters, muscle synergies are extracted from data sample collected in static environment. Such muscle synergies may produce different outcomes in dynamic environment such as different end-effector motion directions, compared with the case in static environment. Although it may be sufficient to utilize more muscle synergies to compensate the effects of the dynamic environment, the applicability of muscle synergies extracted from static environment needs further verification.

Inherent properties of muscle synergies

Muscle synergies that possess minimum control effort properties may not be applicable in achieving control tasks related to force control of muscles. Chapter 4 and chapter 5 propose methods extract muscle synergies from a data sample of control signals that produce end-effector accelerations with minimum control effort in the sense of minimum Euclidean norm. In control tasks related to force control of muscle such as stiffness control of a joint, activating the robot with the minimum muscle activations may not be enough to achieve desired stiffness at joints. Therefore, the muscle synergies inheriting the minimum control effort properties may not be suitable for the force-related control applications.

Scalability

The applicability of the muscle synergies extraction methods presented in chapter 4 and chapter 5 need to be extended and verified in extension to 3D task space application. For the ease of analysis and implementation, the presented methods have been first validated in the application of a human-like robotic arm in 2D task space. More sample points are needed when the problem is extended to 3D task space. As a result, more time and larger memory are required to collect and store the data. Although chapter 4 has presented a sparsification method to handle large-sized data sample, the feasibility of the proposed muscle synergies extraction method in 3D task space is still needed to be examined.

The purposes, contributions and limitations of chapter 3, chapter 4 and chapter 5 are summarized in Figure 6-1.

6.3 Future work

Real robot applications in dynamic environment

All the proposed methods in this research have been verified by simulations in a static environment and 2D task space. Implementation of the proposed methods to real musculoskeletal robots working in 3D task space, and obtaining muscle synergies working in dynamic environments such as interaction with humans and environment

with unexpected disturbances, will be the long-term goals to exploit the compliant and flexible musculoskeletal structure.

Time-invariant vs time-varying synergies

Throughout this research the time-invariant synergies are investigated, which specify the spatial activation pattern of all actuators at a time. In contrast, time-varying synergies have both the spatial component specifying the actuation pattern, and temporal component specifying the time profile of synergy activation. For time-invariant synergies, implementation is easier because synergies can be extracted by common pattern recognition tools [125]. Utilizing time-invariant synergies is equivalent to finding the corresponding temporal profiles of synergy activations. For time-varying synergies, the synergies are more difficult to extract [29]. But utilizing time-varying synergies might further simplify the control problem because it requires much fewer time profile parameters to be determined at the beginning of motion instead of computing solution at each time step. Moreover, time-delay information of a robot is already possessed in the temporal profile of the time-varying synergies. Research of combining the two kinds of synergies is a one of the directions in the future work.

Nonlinear muscles models

The research assumes that the robot is nonlinear in terms of the robot's joint space, but is affine in terms of the control space. However, many musculoskeletal robots are actuated by nonlinear actuators such as pneumatic actuators, which usually have nonlinear control-force relation, or human muscle models having nonlinear dynamics. Extension to handling nonlinear actuators and implementation to real robots will be carried out in the future.

Fully autonomous goal-directed exploration

In the proposed method in chapter 5 the "goals" of exploration are predefined. A similar approach has been adopted in the study of goal babbling process [137]. Incorporating muscle synergies in fully autonomous schemes such as Self-Adaptive Goal Generation Robust Intelligent Adaptive Curiosity algorithm (SAGG-RIAC) [154], where the goals are adaptively generated during exploration, is a potential future work towards the development of fully autonomous musculoskeletal robots.

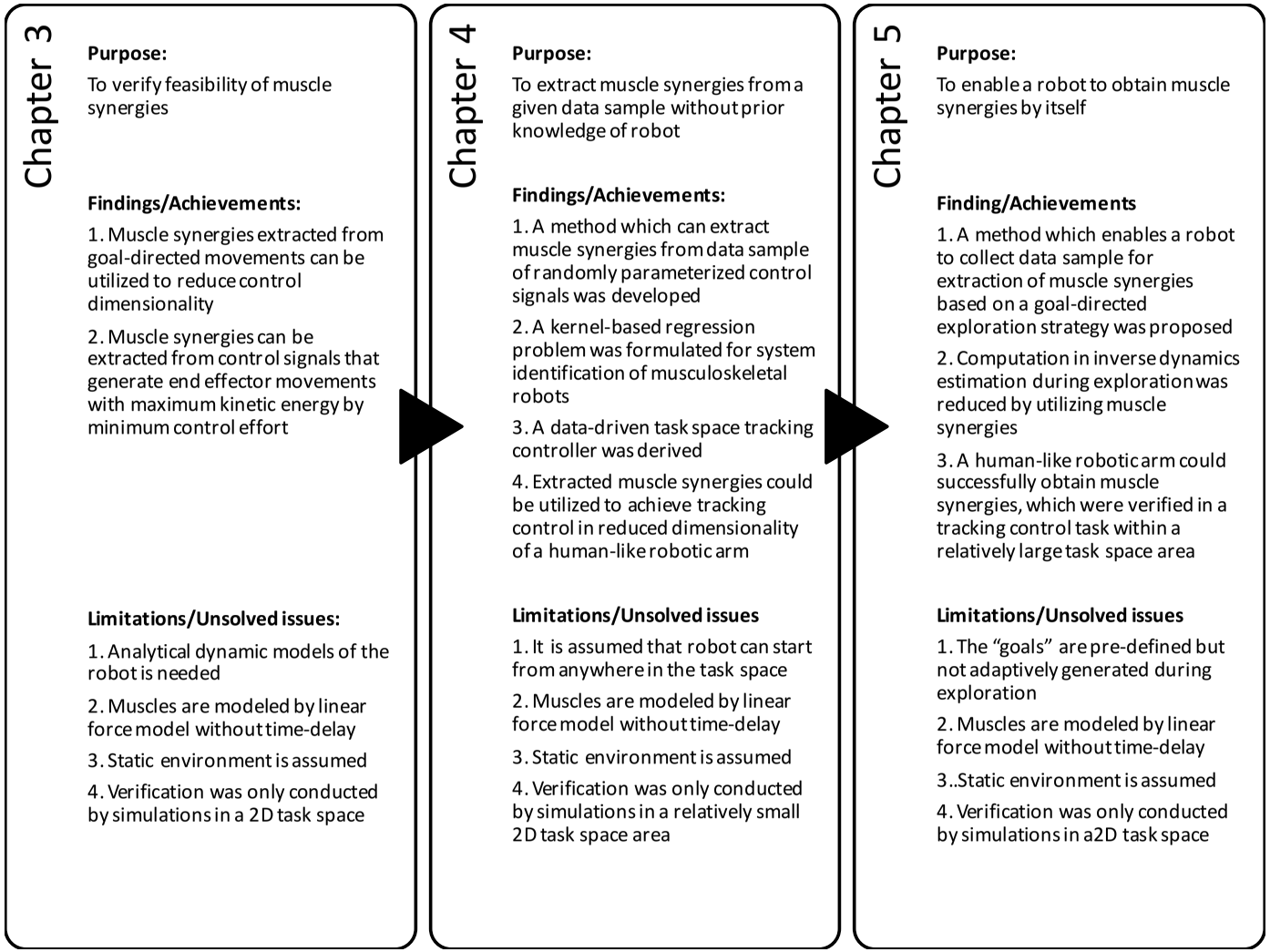


Figure 6-1: Summary of the purposes, contributions and limitations of this research.

Appendix A

A human-like robotic arm simulation platform

A human-like robotic arm simulator was built for the studies in this thesis. Redundancy and high control dimensionality are imparted by 3 joints including a wrist, an elbow and a shoulder, and 10 muscles attached to the arm skeleton. Its dynamics behavior obeys the nonlinear system (2.5) that is linear in control input. For the sake of focusing on investigating the dimensionality reduction functionality of muscle synergies and computational simplicity, the arm is restricted to move in a horizontal plane. The conceptual appearance of the robotic arm is displayed in Fig A-1.

A.1 The musculoskeletal structure

The robotic arm has 3 rigid links: the upper arm, the forearm and the hand. Each link is modeled as a cylinder. The upper arm, the forearm and the hand attach to the shoulder joint, the elbow joint and the wrist joint, respectively. Each joint is revolution joint modeled a massless cylinder.

To represent the position and orientation, a local coordinate frame is attached to each link. The local coordinate frame of a link is attached to the negative side along the longitudinal axis (corresponding to the second element in a 3D position vector) of the cylinder as depicted in Fig. A-2. Each joint is also attached with a coordinate

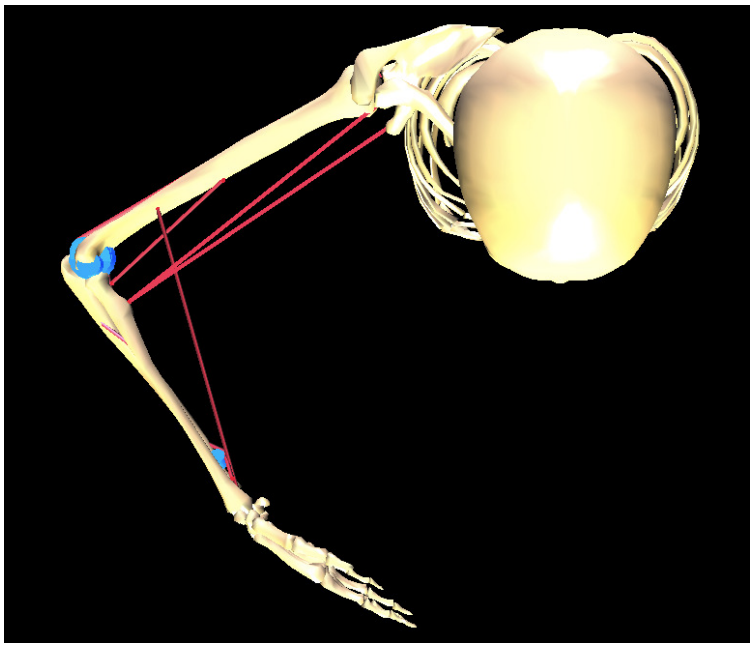


Figure A-1: Conceptual appearance of the human-like robotic arm with 10 muscles.

frame following the Denavit-Hartenberg (DH) convention such that its $\check{z}$-axis is the rotational axis, and the $\check{x}$-axis is the common perpendicular of two successive joints as depicted in Fig. A-3. The origin of the global frame (denoted as $\check{x}_0, \check{y}_0$ in Fig. A-3) of the robotic arm is fixed at the origin of the shoulder joint. The robotic arm is connected by specifying the relative location of a joint cylinder to a link's local coordinate frame, except the case of the should joint is located by a fixed global position.

There are 10 muscles attached to the corresponding links. During motion, muscles are wrapped around the cylindrical surface of the joints. Each muscle produces contractile force $\vec{f}_j, j = 1, ..., 10$ with amplitude that is linearly related to the non-negative, bounded control input $0 \leq u_j \leq 1, j = 1, ..., 10$:

$$\|\vec{f}_j\| = c_j \cdot u_j \tag{A.1}$$

where c_j is the maximum amplitude that the jth muscle can produce. As mentioned in chapter 2, the force direction of a muscle depends on the two attachment points of the muscle and the configuration (joint angles) of the robotic arm. The resulting

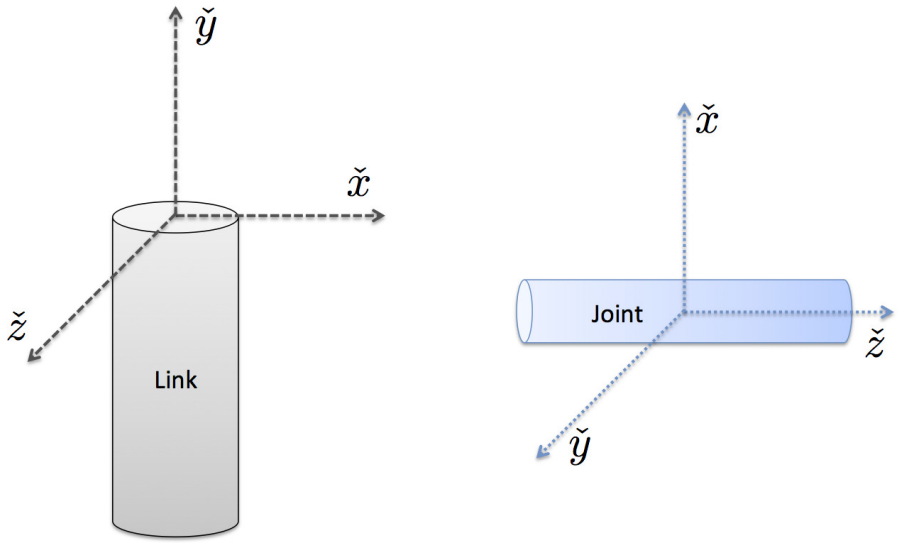


Figure A-2: A link and a joint modeled by cylinders with attached coordinate frames.

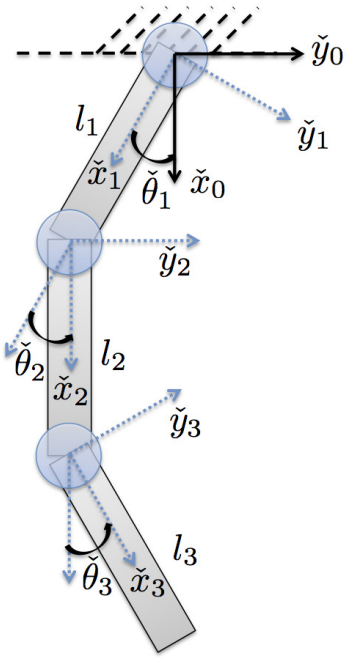


Figure A-3: Coordinate frames attached to the robotic arm using Denavit-Hartenberg convention.

torque at the joints of all the 10 muscles is compute using (2.2).

The physical parameters of the robotic arm are taken reference from anatomical data of an upper arm model in the simulation software MSMS [157]. Table A.1 lists the parameters used in this thesis.

A.2 Kinematics of joints and end effector

By the Denavit-Hartenberg (DH) convention, the global position of a point on the robot can be computed. According to the DH parameters listed in Table A.2 of the robotic arm (Fig. A-3), the homogeneous transformation matrix

$$\mathbf{T}_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i\cos\alpha_{i-1} & \cos\theta_i\cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1}d_i \\ \sin\theta_i\sin\alpha_{i-1} & \cos\theta_i\sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1}d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad i = 1, 2, 3 \quad \text{(A.2)}$$

can be obtained. Given the position vector $\vec{r}_k$ of a point relative to the local coordinate frame of link k, the homogeneous coordinate $\vec{p} = \Big[(\vec{r})^T, 1\Big]^T \in \Re^4$, which is the vector constituted of the global position of a point $\vec{r} \in \Re^3$ and a scalar 1, is evaluated by

$$\vec{p} = \mathbf{T}_1 \cdots \mathbf{T}_m \Big[(\vec{r}_k)^T, 1\Big]^T. \quad \text{(A.3)}$$

A.3 Dynamics model

The dynamics of the robotics arm is derived by Lagrangian equation of motion:

$$\frac{d}{dt}\Big(\frac{\partial L}{\partial \dot{\boldsymbol{x}}}\Big) - \frac{\partial L}{\partial x} = \boldsymbol{F} \quad \text{(A.4)}$$

where $L = K - U$ is the difference of total kinetic energy K and the total potential energy U. $\boldsymbol{x}$ and $\boldsymbol{\tau}$ are the generalized coordinates and the generalized non-conservative

Table A.1: Physical properties of the human-like robotic arm.

Link	Mass (kg)	Moment of inertia (kgm^2)	Length (m)
Upper arm	1.5218	$I_x = 0.0135$ $I_y = 0.0070$ $I_z = 0.0135$	$l_1 = 0.2806$
Forearm	0.8046	$I_x = 0.0047$ $I_y = 0.00161$ $I_z = 0.0047$	$l_2 = 0.241$
Hand	0.4004	$I_x = 0.000754$ $I_y = 0.000316$ $I_z = 0.000615$	$l_3 = 0.1256$

Joint	Joint location (local coordinate frame)	Cylinder radius (m)
Shoulder	$[-0.05511, -0.0096, 0.1374]^T$ (Global)	0.01
Elbow	$[0, -0.2806, 0]^T$ (Upper arm)	0.01
Hand	$[0, -0.241, 0]^T$ (Forearm)	0.01

Muscle	Attachment points (local coordinate frame)	Maximum force (N)
BIC_{ln}	$[0.04989, -0.00429, 0.11332]^T$ (Global) $[0.01364, -0.03203, 8.7e^{-4}]^T$ (Forearm)	312.1
BIC_{sh}	$[0.02531, -0.00878, 0.11265]^T$ (Global) $[0.01355, -0.03203, 8.9e^{-4}]^T$ (Forearm)	253.2
BRA	$[0.0254, -0.14083, 0.00771]^T$ (Upper arm) $[0.00606, -0.02557, 0.00216]^T$ (Forearm)	621.4
BRD	$[-0.00742, -0.20239, 0.00833]^T$ (Upper arm) $[0.00311, -0.22186, 6.2e^{-4}]^T$ (Forearm)	133.3
PT	$[-0.0141; -0.27551; -0.019476]^T$ (Upper arm) $[0.0017, -0.09916, 0.01379]^T$ (Forearm)	480
TRI_{ln}	$[-0.07363, -0.02957, 0.115]^T$ (Global) $[-0.02266, -0.01411, 0.00175]^T$ (Forearm)	453.4
TRI_{lt}	$[-0.01424, -0.07357, 0.00589]^T$ (Upper arm) $[-0.02341, -0.01397, 0.00104]^T$ (Forearm)	1127.4
TRI_{m}	$[-0.01679, -0.20597, 9.2e^{-4}]^T$ (Upper arm) $[-0.02272, -0.01423, 0.00161]^T$ (Forearm)	581.9
$Hand_{f}$	$[0.005, -0.05, 0]^T$ (Forearm) $[0.005, -0.05, 0]^T$ (Hand)	300
$Hand_{e}$	$[-0.005, -0.05, 0]^T$ (Forearm) $[-0.005, -0.05, 0]^T$ (Hand)	300

Table A.2: Denavit-Hartenberg parameters of the robotic arm for computation of the end effector position.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	$\breve{\theta}_1$
2	0	l_1	0	$\breve{\theta}_2$
3	0	l_2	0	$\breve{\theta}_3$
4	0	l_3	0	0

forces exerted to the robotic arm. In our case, $\boldsymbol{x}$ is the joint angles. The generalized forces $\boldsymbol{F}$ consists of the resulting torque $\boldsymbol{\tau}$ at the joints provided by the muscle forces $\vec{f}_j, j = 1, \ldots, 10$ and the frictional force $-b\dot{\boldsymbol{x}}$ at the joints:

$$\boldsymbol{F} = \boldsymbol{\tau} - b\dot{\boldsymbol{x}} \tag{A.5}$$

wher $b \in \Re^1$ is the frictional coefficient. The torque $\boldsymbol{\tau}$ is computed by (2.2) in chapter 2. Since the motion is constrained on the horizontal plane, the potential energy is zero $U = 0$. The kinetic engergy of each link is computed by

$$K_i = \frac{1}{2} m_i \|\dot{\vec{r}}^{c_i}\|^2 + \frac{1}{2} I_i \| \sum_{j=1}^{i} \dot{x}_i \|^2 \tag{A.6}$$

where m_i and I_i are the mass and the moment of inertia at the center of mass of the link i. $\vec{r}^{c_i}$ is the global position of the center of mass of link i. The center of mass is assumed at the mid-point of each link. The resulting robot's dynamics is

$$M\Big(\boldsymbol{x}(t)\Big)\ddot{\boldsymbol{x}}(t) + C\Big(\boldsymbol{x}(t), \dot{\boldsymbol{x}}(t)\Big)\dot{\boldsymbol{x}}(t) + G\Big(\boldsymbol{x}(t)\Big) = \boldsymbol{\tau}\Big(\boldsymbol{x}(t), \boldsymbol{u}(t), t\Big) \tag{A.7}$$

where $\boldsymbol{x}(t) \in \Re^{n^x}$, $\dot{\boldsymbol{x}}(t) \in \Re^{n^x}$ and $\ddot{\boldsymbol{x}}(t) \in \Re^{n^x}$ are the joint angles, joint velocities and joint accelerations, respectively.

Integration is carried out by Euler integration. Time step 0.001 second was used in the simulation in chapter 3. Time step 0.01 second was used in the simulation in both chapter 4 and chapter 5. The simulation platform was implemented using MATLAB.

Bibliography

[1] M. Nordin and V. H. Frankel, *Basic biomechanics of the musculoskeletal system*. Lippincott Williams & Wilkins, 2001.

[2] B. Tondu and P. Lopez, "Modeling and control of mckibben artificial muscle robot actuators," *IEEE Control Systems*, vol. 20, no. 2, pp. 15–38, 2000.

[3] Y. Nakata, A. Ide, Y. Nakamura, K. Hirata, and H. Ishiguro, "Hopping of a monopedal robot with a biarticular muscle driven by electromagnetic linear actuators," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 3153–3160.

[4] M. Vande Weghe, M. Rogers, M. Weissert, and Y. Matsuoka, "The act hand: design of the skeletal structure," in *International Conference on Robotics and Automation (ICRA)*, vol. 4. IEEE, 2004, pp. 3375–3379.

[5] R. Blickhan, "The spring-mass model for running and hopping," *Journal of biomechanics*, vol. 22, no. 11, pp. 1217–1227, 1989.

[6] A. Verl, A. Albu-Schäffer, O. Brock, and A. Raatz, *Soft Robotics: Transferring Theory to Application*. Springer, 2015.

[7] V. Salvucci, Y. Kimura, S. Oh, and Y. Hori, "Force maximization of biarticularly actuated manipulators using infinity norm," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 3, pp. 1080–1089, 2013.

[8] N. G. Tsagarakis and D. G. Caldwell, "Development and control of a 'soft-actuated' exoskeleton for use in physiotherapy and training," *Autonomous Robots*, vol. 15, no. 1, pp. 21–33, 2003.

[9] P. K. Artemiadis and K. J. Kyriakopoulos, "Emg-based control of a robot arm using low-dimensional embeddings," *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 393–398, 2010.

[10] P. Artemiadis, "Emg-based robot control interfaces: Past, present and future," *Advances in Robotics & Automation*, vol. 1, no. 02, pp. 10–12, 2012.

[11] Z. Xu and T. Emanuel, "Design of a highly biomimetic anthropomorphic robotic hand towards artificial limb regeneration," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016, p. (to be appeared).

[12] L. H. Ting, "Dimensional reduction in sensorimotor systems: a framework for understanding muscle coordination of posture," *Progress in brain research*, vol. 165, pp. 299–321, 2007.

[13] N. A. Bernstein, *The co-ordination and regulation of movements*. Pergamon Press Ltd., 1967.

[14] M. L. Latash, J. P. Scholz, and G. Schoner, "Toward a new theory of motor synergies," *Motor Control*, vol. 11, no. 3, p. 276, 2007.

[15] T. Flash and B. Hochner, "Motor primitives in vertebrates and invertebrates," *Current opinion in neurobiology*, vol. 15, pp. 660–666, 2005.

[16] W. A. Lee, "Neuromotor synergies as a basis for coordinated intentional action," *Journal of motor behavior*, vol. 16, no. 2, pp. 135–170, 1984.

[17] A. d'Avella, P. Saltiel, and E. Bizzi, "Combinations of muscle synergies in the construction of a natural motor behavior," *Nature neuroscience*, vol. 6, no. 3, pp. 300–308, 2003.

[18] S. Yakovenko, N. Krouchev, and T. Drew, "Sequential activation of motor cortical neurons contributes to intralimb coordination during reaching in the cat by modulating muscle synergies," *Journal of neurophysiology*, vol. 105, no. 1, pp. 388–409, 2011.

[19] S. A. Overduin, A. d'Avella, J. Roh, and E. Bizzi, "Modulation of muscle synergy recruitment in primate grasping," *The Journal of Neuroscience*, vol. 28, no. 4, pp. 880–892, 2008.

[20] C. Alessandro, I. Delis, F. Nori, S. Panzeri, and B. Berret, "Muscle synergies in neuroscience and robotics: from input-space to task-space perspectives," *Frontiers in Computational Neuroscience*, vol. 7, no. 43, 2013.

[21] M. L. Latash, J. P. Scholz, and G. Schöner, "Motor control strategies revealed in the structure of motor variability," *Exercise and sport sciences reviews*, vol. 30, no. 1, pp. 26–31, 2002.

[22] M. L. Latash, "Motor synergies and the equilibrium-point hypothesis," *Motor control*, vol. 14, no. 3, p. 294, 2010.

[23] M. C. Tresch, P. Saltiel, and E. Bizzi, "The construction of movement by the spinal cord," *Nature neuroscience*, vol. 2, pp. 162–167, 1999.

[24] A. d'Avella and E. Bizzi, "Shared and specific muscle synergies in natural motor behaviors," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 8, pp. 3076–3081, 2005.

[25] L. H. Ting and J. M. Macpherson, "A limited set of muscle synergies for force control during a postural task," *Journal of neurophysiology*, vol. 93, no. 1, pp. 609–613, 2005.

[26] G. Torres-Oviedo, J. M. Macpherson, and L. H. Ting, "Muscle synergy organization is robust across a variety of postural perturbations," *Journal of neurophysiology*, vol. 96, no. 3, pp. 1530–1546, 2006.

[27] T. Brochier, R. L. Spinks, M. A. Umilta, and R. N. Lemon, "Patterns of muscle activity underlying object-specific grasp by the macaque monkey," *Journal of Neurophysiology*, vol. 92, no. 3, pp. 1770–1782, 2004.

[28] A. d'Avella, A. Portone, L. Fernandez, and F. Lacquaniti, "Control of fast-reaching movements by muscle synergy combinations," *The Journal of neuroscience*, vol. 26, no. 30, pp. 7791–7810, 2006.

[29] A. d'Avella, L. Fernandez, A. Portone, and F. Lacquaniti, "Modulation of phasic and tonic muscle synergies with reaching direction and speed," *Journal of Neurophysiology*, vol. 100, pp. 1433–1454, 2008.

[30] J. Roh, W. Z. Rymer, and R. F. Beer, "Robustness of muscle synergies underlying three-dimensional force generation at the hand in healthy humans," *Journal of neurophysiology*, vol. 107, no. 8, pp. 2123–2142, 2012.

[31] Y. P. Ivanenko, R. Grasso, M. Zago, M. Molinari, G. Scivoletto, V. Castellano, V. Macellari, and F. Lacquaniti, "Temporal components of the motor patterns expressed by the human spinal cord reflect foot kinematics," *Journal of neurophysiology*, vol. 90, no. 5, pp. 3555–3565, 2003.

[32] Y. P. Ivanenko, R. E. Poppele, and F. Lacquaniti, "Five basic muscle activation patterns account for muscle activity during human locomotion," *The Journal of physiology*, vol. 556, no. 1, pp. 267–282, 2004.

[33] F. J. Valero-Cuevas, M. Venkadesan, and E. Todorov, "Structured variability of muscle activations supports the minimal intervention principle of motor control," *Journal of Neurophysiology*, vol. 102, no. 1, pp. 59–68, 2009.

[34] J. J. Kutch and F. J. Valero-Cuevas, "Challenges and new approaches to proving the existence of muscle synergies of neural origin," *PLoS Computational Biology*, vol. 8, no. 5, p. e1002434, 2012.

[35] R. Holdefer and L. Miller, "Primary motor cortical neurons encode functional muscle synergies," *Experimental Brain Research*, vol. 146, no. 2, pp. 233–243, 2002.

[36] V. C. Cheung, L. Piron, M. Agostini, S. Silvoni, A. Turolla, and E. Bizzi, "Stability of muscle synergies for voluntary actions after cortical stroke in humans," *Proceedings of the National Academy of Sciences*, vol. 106, no. 46, pp. 19 563–19 568, 2009.

[37] V. C. Cheung, A. Turolla, M. Agostini, S. Silvoni, C. Bennis, P. Kasi, S. Paganoni, P. Bonato, and E. Bizzi, "Muscle synergy patterns as physiological markers of motor cortical damage," *Proceedings of the National Academy of Sciences*, vol. 109, no. 36, pp. 14 652–14 656, 2012.

[38] L. Gizzi, J. F. Nielsen, F. Felici, Y. P. Ivanenko, and D. Farina, "Impulses of activation but not motor modules are preserved in the locomotion of subacute stroke patients," *Journal of neurophysiology*, vol. 106, no. 1, pp. 202–210, 2011.

[39] H. Broer and F. Takens, *Dynamical systems and chaos*. Springer Science & Business Media, 2010, vol. 172.

[40] M. Berniker, A. Jarc, E. Bizzi, and M. C. Tresch, "Simplified and effective motor control based on muscle synergies to exploit musculoskeletal dynamics," *PNAS*, vol. 106, pp. 7601–7606, 2009.

[41] R. R. Neptune, D. J. Clark, and S. A. Kautz, "Modular control of human walking: a simulation study," *Journal of biomechanics*, vol. 42, no. 9, pp. 1282–1287, 2009.

[42] J. L. Allen and R. R. Neptune, "Three-dimensional modular control of human walking," *Journal of biomechanics*, vol. 45, no. 12, pp. 2157–2163, 2012.

[43] M. Kawato, "Internal models for motor control and trajectory planning," *Current opinion in neurobiology*, vol. 9, no. 6, pp. 718–727, 1999.

[44] M. Kawato, K. Furukawa, and R. Suzuki, "A hierarchical neural-network model for control and learning of voluntary movement," *Biological cybernetics*, vol. 57, no. 3, pp. 169–185, 1987.

[45] M. I. Jordan, "Computational aspects of motor control and motor learning," *Handbook of perception and action: motor skills*, vol. 2, pp. 71–118, 1996.

[46] R. C. Miall and D. M. Wolpert, "Forward models for physiological motor control," *Neural networks*, vol. 9, no. 8, pp. 1265–1279, 1996.

[47] D. M. Wolpert and J. R. Flanagan, "Motor prediction," *Current Biology*, vol. 11, no. 18, pp. R729–R732, 2001.

[48] R. R. Neptune, C. P. McGowan, and S. A. Kautz, "Forward dynamics simulations provide insight into muscle mechanical work during human locomotion," *Exercise and sport sciences reviews*, vol. 37, no. 4, p. 203, 2009.

[49] D. M. Wolpert and Z. Ghahramani, "Computational principles of movement neuroscience," *nature neuroscience*, vol. 3, pp. 1212–1217, 2000.

[50] D. M. Wolpert, K. Doya, and M. Kawato, "A unifying computational framework for motor control and social interaction," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 358, no. 1431, pp. 593–602, 2003.

[51] P. N. Sabes, "The planning and control of reaching movements," *Current opinion in neurobiology*, vol. 10, no. 6, pp. 740–746, 2000.

[52] A. d'Avella and F. Lacquaniti, "Control of reaching movements by muscle synergy combinations," *Front. Comput. Neurosci*, vol. 7, no. 42, pp. 10–3389, 2013.

[53] A. d'Avella, A. Portone, and F. Lacquaniti, "Superposition and modulation of muscle synergies for reaching in response to a change in target location," *Journal of neurophysiology*, vol. 106, no. 6, pp. 2796–2812, 2011.

[54] M. Graziano, *The Intelligent Movement Machine: An Ethological Perspective on the Primate Motor System.* Oxford University Press, 2008.

[55] W. J. Kargo and S. F. Giszter, "Rapid correction of aimed movements by summation of force-field primitives," *The Journal of Neuroscience*, vol. 20, no. 1, pp. 409–426, 2000.

[56] J. L. McKay and L. H. Ting, "Functional muscle synergies constrain force production during postural tasks," *Journal of biomechanics*, vol. 41, no. 2, pp. 299–306, 2008.

[57] C. G. Atkeson and J. M. Hollerbach, "Kinematic features of unrestrained vertical arm movements," *The Journal of Neuroscience*, vol. 5, no. 9, pp. 2318–2330, 1985.

[58] L. H. Snyder, "Coordinate transformations for eye and arm movements in the brain," *Current opinion in neurobiology*, vol. 10, no. 6, pp. 747–754, 2000.

[59] F. Lacquaniti, "Central representations of human limb movement as revealed by studies of drawing and handwriting," *Trends in neurosciences*, vol. 12, no. 8, pp. 287–291, 1989.

[60] E. A. Henis and T. Flash, "Mechanisms underlying the generation of averaged modified trajectories," *Biological Cybernetics*, vol. 72, no. 5, pp. 407–419, 1995.

[61] R. A. Andersen, G. K. Essick, and R. M. Siegel, "Encoding of spatial location by posterior parietal neurons," *Science*, vol. 230, no. 4724, pp. 456–458, 1985.

[62] P. Haggard, K. Hutchinson, and J. Stein, "Patterns of coordinated multi-joint movement," *Experimental Brain Research*, vol. 107, no. 2, pp. 254–266, 1995.

[63] J. Martin, S. Cooper, and C. Ghez, "Kinematic analysis of reaching in the cat," *Experimental brain research*, vol. 102, no. 3, pp. 379–392, 1995.

[64] A. d'Avella, L. Fernandez, A. Portone, and F. Lacquaniti, "Modulation of phasic and tonic muscle synergies with reaching direction and speed," *Journal of neurophysiology*, vol. 100, no. 3, pp. 1433–1454, 2008.

[65] S. A. Chvatal, G. Torres-Oviedo, S. A. Safavynia, and L. H. Ting, "Common muscle synergies for control of center of mass and force in nonstepping and stepping postural behaviors," *Journal of neurophysiology*, vol. 106, no. 2, pp. 999–1015, 2011.

[66] E. Todorov, "Optimality principles in sensorimotor control," *Nature neuroscience*, vol. 7, no. 9, pp. 907–915, 2004.

[67] E. Todorov, W. Li, and X. Pan, "From task parameters to motor synergies: A hierarchical framework for approximately optimal control of redundant manipulators," *Journal of robotic systems*, vol. 22, no. 11, pp. 691–710, 2005.

[68] E. Todorov and M. I. Jordan, "Optimal feedback control as a theory of motor coordination," *Nature Neuroscience*, vol. 5, no. 11, pp. 1226–1235, 2002.

[69] E. Todorov, W. Li, and X. Pan, "From task parameters to motor synergies: A hierarchical framework for approximately optimal control of redundant manipulators," *Journal of robotic systems*, vol. 22, pp. 691–710, 2005.

[70] M. Chhabra and R. A. Jacobs, "Properties of synergies arising from a theory of optimal motor behavior," *Neural computation*, vol. 18, no. 10, pp. 2320–2342, 2006.

[71] M. C. Tresch and A. Jarc, "The case for and against muscle synergies," *Current opinion in neurobiology*, vol. 19, no. 6, pp. 601–607, 2009.

[72] J. L. McKay and L. H. Ting, "Optimization of muscle activity for task-level goals predicts complex changes in limb forces across biomechanical contexts," *PLoS Comput Biol*, vol. 8, no. 4, p. e1002465, 2012.

[73] R. F. Stengel, *Optimal control and estimation*. Courier Corporation, 2012.

[74] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal Control*. John Wiley & Sons, 2012.

[75] E. Todorov, "Optimal control theory," *Bayesian brain: probabilistic approaches to neural coding*, pp. 269–298, 2006.

[76] J. J. Craig, *Introduction to robotics: mechanics and control*. Pearson Prentice Hall Upper Saddle River, 2005, vol. 3.

[77] J.-J. E. Slotine, W. Li *et al.*, *Applied nonlinear control*. Prentice-hall Englewood Cliffs, NJ, 1991, vol. 199, no. 1.

[78] D. Liu and E. Todorov, "Hierarchical optimal control of a 7-dof arm model," in *Adaptive Dynamic Programming and Reinforcement Learning, 2009. ADPRL'09. IEEE Symposium on*. IEEE, 2009, pp. 50–57.

[79] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," *The journal of Neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.

[80] Y. Uno, M. Kawato, and R. Suzuki, "Formation and control of optimal trajectory in human multijoint arm movement," *Biological cybernetics*, vol. 61, no. 2, pp. 89–101, 1989.

[81] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *Robotics and Automation, IEEE Journal of*, vol. 3, no. 1, pp. 43–53, 1987.

[82] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal, "Operational space control: A theoretical and empirical comparison," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 737–757, 2008.

[83] J. Peters, M. Mistry, F. Udwadia, J. Nakanishi, and S. Schaal, "A unifying framework for robot control with redundant dofs," *Autonomous Robots*, vol. 24, no. 1, pp. 1–12, 2008.

[84] L. Sentis and O. Khatib, "Synthesis of whole-body behaviors through hierarchical control of behavioral primitives," *International Journal of Humanoid Robotics*, vol. 2, no. 04, pp. 505–518, 2005.

[85] J. Peters and S. Schaal, "Learning to control in operational space," *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 197–212, 2008.

[86] D. Nguyen-Tuong and J. Peters, "Online kernel-based learning for task-space tracking robot control," *Neural Networks and Learning Systems, IEEE Transactions on*, vol. 23, no. 9, pp. 1417–1425, 2012.

[87] M. W. Spong and M. Vidyasagar, *Robot dynamics and control*. John Wiley & Sons, 2008.

[88] K. S. Narendra and A. M. Annaswamy, *Stable adaptive systems*. Courier Corporation, 2012.

[89] S. Nicosia and P. Tomei, "Model reference adaptive control algorithms for industrial robots," *Automatica*, vol. 20, no. 5, pp. 635–644, 1984.

[90] J. M. Maciejowski, *Predictive control: with constraints*. Pearson education, 2002.

[91] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.

[92] F. A. Mussa-Ivaldi, "Nonlinear force fields: a distributed system of control primitives for representing and learning movements," in *Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on*. IEEE, 1997, pp. 84–90.

[93] E. Bizzi, N. Accornero, W. Chapple, and N. Hogan, "Posture control and trajectory formation during arm movement," *The Journal of Neuroscience*, vol. 4, no. 11, pp. 2738–2744, 1984.

[94] N. Hogan, "An organizing principle for a class of voluntary movements," *The Journal of Neuroscience*, vol. 4, no. 11, pp. 2745–2754, 1984.

[95] F. Nori and R. Frezza, "A control theory approach to the analysis and synthesis of the experimentally observed motion primitives," *Biological cybernetics*, vol. 93, no. 5, pp. 323–342, 2005.

[96] M. Chhabra and R. A. Jacobs, "Properties of synergies arising from a theory of optimal motor behavior," *Neural Computation*, vol. 18, pp. 2320–2342, 2006.

[97] E. Todorov, "Compositionality of optimal control laws," in *Advances in Neural Information Processing Systems*, 2009, pp. 1856–1864.

[98] E. Todorov and Z. Ghahramani, "Unsupervised learning of sensory-motor primitives," in *Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE*, vol. 2. IEEE, 2003, pp. 1750–1753.

[99] R. S. Sutton, A. G. Barto, and R. J. Williams, "Reinforcement learning is direct adaptive optimal control," *Control Systems, IEEE*, vol. 12, no. 2, pp. 19–22, 1992.

[100] M. Malhotra, E. Rombokas, E. Theodorou, E. Todorov, and Y. Matsuoka, "Reduced dimensionality control for the act hand," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 5117–5122.

[101] H. G. Marques, P. Schaffner, and N. Kuppuswamy, "Unsupervised learning of a reduced dimensional controller for a tendon driven robot platform," in *From Animals to Animats 12*. Springer, 2012, pp. 351–360.

[102] M. Millard, T. Uchida, A. Seth, and S. L. Delp, "Flexing computational muscle: Modeling and simulation of musculotendon dynamics," *ASME Journal of Biomechanical Engineering*, vol. 135, no. 2, p. 021105, 2013.

[103] F. E. Zajac, "Muscle and tendon: Properties, models, scaling, and application to biomechanics and motor control," *Critical Reviews in Biomedical Engineering*, vol. 17, no. 4, pp. 359–411, 1989.

[104] J. Winters, "Hill-based muscle models: a systems engineering perspective," *Multiple Muscle Systems: Biomechanics and Movement Organization*, pp. 69–91, 1990.

[105] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, "Algorithms and applications for approximate nonnegative matrix factorization," *Computational statistics & data analysis*, vol. 52, no. 1, pp. 155–173, 2007.

[106] G. Torres-Oviedo and L. H. Ting, "Muscle synergies characterizing human postural responses," *Journal of neurophysiology*, vol. 98, no. 4, pp. 2144–2156, 2007.

[107] V. Krishnamoorthy, M. L. Latash, J. P. Scholz, and V. M. Zatsiorsky, "Muscle synergies during shifts of the center of pressure by standing persons," *Experimental brain research*, vol. 152, no. 3, pp. 281–292, 2003.

[108] C. M. Bishop, *Pattern Recongition and Machine Learning*. Cambridge, U.K.: Springer, 2006.

[109] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural computation*, vol. 7, no. 6, pp. 1129–1159, 1995.

[110] A. Ajiboye and R. Weir, "Muscle synergies as a predictive framework for the emg patterns of new hand postures," *Journal of neural engineering*, vol. 6, no. 3, p. 036004, 2009.

[111] A. M. Sabatini, "Identification of neuromuscular synergies in natural upper-arm movements," *Biological cybernetics*, vol. 86, no. 4, pp. 253–262, 2002.

[112] A. E. Bryson, *Dynamic optimization*. Addison Wesley Longman Menlo Park, 1999.

[113] J. Betts and I. Kolmanovsky, "Practical methods for optimal control using nonlinear programming," *Applied Mechanics Reviews*, vol. 55, p. 68, 2002.

[114] F. L. Lewis and V. L. Syrmos, *Optimal control*. John Wiley & Sons, 1995.

[115] P. Hsu, J. Mauser, and S. Sastry, "Dynamic control of redundant manipulators," *Journal of Robotic Systems*, vol. 6, no. 2, pp. 133–148, 1989.

[116] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal, "Operational space control: A theoretical and empirical comparison," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 737–757, June 2008.

[117] R. A. Schmidt and T. Lee, *Motor Control and Learning: A Behavioral Emphasis*, 3rd ed. Human Kinetics, 1999.

[118] M. L. Latash, J. P. Scholz, and G. Schöner, "Motor control strategies revealed in the structure of motor variability," *Exercise and Sport Sciences Reviews*, vol. 30, pp. 26–31, 2002.

[119] M. C. Tresch and A. Jarc, "The case for and against muscle synergies," *Current Opinion in Neurobiology*, vol. 19, pp. 601–607, 2009.

[120] C. M. Harris and D. M. Wolpert, "Signal-dependent noise determines motor planning," *Nature*, vol. 394, pp. 780–784, 1998.

[121] E. Todorov, "Optimality principles in sensorimotor control," *Nature Neuroscience*, vol. 7, no. 9, pp. 907–915, 2004.

[122] R. E. Bellman, *Dynamic Programming*. Courier Dover Publications, 2003.

[123] W. Li and E. Todorov, "Iterative linearization methods for approximately optimal control and estimation of non-linear stochastic system," *International Journal of Control*, vol. 80, no. 9, pp. 1439–1453, 2007.

[124] T. Flash and N. Hogan, "The coordination of arm movements: An experimentally confirmed mathematical model," *The Journal of Neuroscience*, vol. 5, pp. 1688–1703, Jul. 1985.

[125] M. C. Tresch, V. C. K. Cheung, and A. d'Avella, "Matrix factorization algorithms for the identification of muscle synergies: Evaluation on simulated and experimental data sets," *Journal of Neurophysiology*, vol. 95, pp. 2199–2212, 2006.

[126] S. Schaal and N. Schweighofer, "Computational motor control in humans and robots," *Current Opinion in Neurobiology*, vol. 15, pp. 675–682, 2005.

[127] J. Romero, T. Feix, C. Henrik Ek, H. Kjellström, and D. Kragic, "Extracting postural synergies for robotic grasping," *IEEE Transactions on Robotics*, vol. 29, no. 6, pp. 1342–1352, Decemeber 2013.

[128] A. Diamond and O. E. Hollan, "Reaching control of a full-torso, modelled musculoskeletal robot using muscle synergies emergent under reinforcement learning," *Bioinspiration & Biomimetics*, vol. 9, no. 1, p. 016015, March 2014.

[129] K. S. Luck, G. Neumann, E. Berger, J. Peters, and H. B. Amor, "Latent space policy search for robotics," in *IEEE International Conference on Intelligent Robots and Systems, IROS*, 2014, pp. 1434–1440.

[130] J. Peters and S. Schaal, "Learning to control in operational space," *International Journal of Robotics Research*, vol. 27, no. 2, pp. 197–212, 2008.

[131] D. Nguyen-Tuong and J. Peters, "Online kernel-based learning for task-space tracking robot control," *IEEE Transactions on Neural Networks*, vol. 23, no. 9, pp. 1417–1425, 2012.

[132] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001.

[133] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An introduction to kernel-based learning algorithms," *Neural Networks, IEEE Transactions on*, vol. 12, no. 2, pp. 181–201, 2001.

[134] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, New Jersey: Prentice Hall, 1991.

[135] V. Sankaranarayanan and A. D. Mahindrakar, "Control of a class of underactuated mechanical systems using sliding modes," *IEEE Transactions On Robotics*, vol. 25, no. 2, pp. 459–467, 2009.

[136] D. Caligiore, T. Ferrauto, D. Parisi, N. Accornero, M. Capozza, and G. Baldassarre, "Using motor babbling and hebb rules for modeling the development of reaching with obstacles and grasping," in *International Conference on Cognitive Systems*, 2008, pp. E1–8.

[137] M. Rolf, J. J. Steil, and M. Gienger, "Goal babbling permits direct learning of inverse kinematics," *IEEE Transactions on Autonomous Mental Development*, vol. 3, no. 2, pp. 216–229, September 2010.

[138] C. Salaün, V. Padois, and O. Sigaud, "Control of redundant robots using learned models: an operational space control approach," in *Proceedings of the 2009 IEEE International Conference on Intelligent Robots and Systems*, 2009.

[139] M. E. Tipping. [Online]. Available: http://www.miketipping.com/downloads.htm

[140] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *The journal of machine learning research*, vol. 1, pp. 211–244, 2001.

[141] A. C. F. Michael E. Tipping, "Fast marginal likelihood maximisation for sparse bayesian models," in *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, 2003, pp. 3–6.

[142] C. Alessandro, J. P. Carbajal, and A. d'Avella, "A computational analysis of motor synergies by dynamic response decomposition," *Frontiers in Computational Neuroscience*, vol. 7, no. 191, 2014.

[143] M. L. Latash, "Stages in learning motor synergies: A view based on the equilibrium-point hypothesis," *Human Movement Science*, vol. 29, no. 5, pp. 642–654, 2010.

[144] M. Malhotra, E. Rombokas, E. Theodorou, T. Emanuel, and Y. Matsuoka, "Reduced dimensionality control for the act hand," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 5117–5122.

[145] S. Bitzer and S. Vijayakumar, "Latent spaces for dynamic movement primitives," in *IEEE-RAS International Conference on Humanoid Robots*, 2009, pp. 574–581.

[146] M. C. Tresch, V. C. Cheung, and A. d'Avella, "Matrix factorization algorithms for the identification of muscle synergies: evaluation on simulated and experimental data sets," *Journal of Neurophysiology*, vol. 95, no. 4, pp. 2199–2212, 2006.

[147] M. Maier and M. Hepp-Reymond, "Emg activation patterns during force production in precision grip. ii. muscular synergies in the spatial and temporal domain." *Experimental brain research*, vol. 103, no. 1, pp. 123–136, 1994.

[148] G. Torres-Oviedo and L. H. Ting, "Subject-specific muscle synergies in human balance control are consistent across different biomechanical contexts," *Journal of neurophysiology*, vol. 103, no. 6, pp. 3084–3098, 2010.

[149] F. De Groote, I. Jonkers, and J. Duysens, "Task constraints and minimization of muscle effort result in a small number of muscle synergies during gait," *Frontiers in Computational Neuroscience*, vol. 8, no. 115, 2014.

[150] C. von Hofsten, "An action perspective on motor development," *Trends in cognitive sciences*, vol. 8, no. 6, pp. 266–272, 2004.

[151] A. L. van der Meer, "Keeping the arm in the limelight: Advanced visual control of arm movements in neonates," *European Journal of Paediatric Neurology*, vol. 1, no. 4, pp. 103–108, 1997.

[152] C. von Hofsten, "Eye-hand coordination in the newborn," *Developmental Psychology*, vol. 18, no. 3, pp. 450–461, 1982.

[153] M. Rolf and J. J. Steil, "Efficient Exploratory Learning of Inverse Kinematics on a Bionic Elephant Trunk," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 6, pp. 1147–1160, 2014.

[154] A. Baranes and P.-Y. Oudeyer, "Active learning of inverse models with intrinsically motivated goal exploration in robots," *Robotics and Autonomous Systems*, vol. 61, no. 1, pp. 49–73, 2013.

[155] C. Hartmann, J. Boedecker, O. Obst, S. Ikemoto, and M. Asada, "Real-time inverse dynamics learning for musculoskeletal robots based on echo state gaussian process regression," in *Robotics: Science and Systems*, 2012, pp. 1–8.

[156] E. N. Berthier, R. K. Clifton, D. D. McCall, and D. J. Robin, "Proximodistal structure of early reaching in human infants," *Experimental Brain Research*, vol. 127, no. 3, pp. 259–269, 1999.

[157] R. Davoodi and G. E. Loeb, "Msms software for vr simulations of neural prostheses and patient training and rehabilitation," *Studies in Health Technology and Informatics*, vol. 163, pp. 156–162, 2011.

Acknowledgments

I would like to thank Professor Hiroshi Ishiguro, providing me a chance to pursue the Ph.D. degree at the Osaka University. I would like to thank Professor Hiroshi Ishiguro, Dr. Yutaka Nakamura, Dr. Tomoyuki Yamamoto, Dr. Fabio Dalla Libera and Dr. Yoshihiro Nataka for their supervision. The present work would not have been possible without their invaluable supports, guidance and technical knowledge. Thank you for their patience and efforts that always generously sharing their valuable knowledge over all these years. Many thanks to Professor Koh Hosoda and Professor Tatsuo Arai for taking their time to participate as committee members.

I also would like to thank my colleagues in the Ishiguro's Lab. They always stay around me in the school and in the private time. Without their smiles and their valuable supports, I cannot overcome many harsh moments and complete the works in this thesis.

Finally, I would like to thank my friends and families. They are always my motivation to finish this thesis. This belongs to everyone who supports me.

Dedicated to my beloved families

Related publications

The following are the publications related to this thesis.

1. K. C. D. Fu, Y. Nakamura, T. Yamamoto, and H. Ishiguro, "Studies of motor synergies in generating optimal goal-directed movements in human-like robotic arm," in *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2012, pp. 808–813. (Chapter 3)

2. K. C. D. Fu, Y. Nakamura, T. Yamamoto, and H. Ishiguro, "Analysis of motor synergies utilization for optimal movement generation for a human-like robotic arm," *International Journal of Automation and Computing*, vol. 10, pp, 515–524, 2013. (Chapter 3)

3. K. C. D. Fu, F. D. Libera, and H. Ishiguro, "Extracting motor synergies from random movements for low-dimensional task-space control of musculoskeletal robots," *Bioinspiration & Biomimetics*, vol. 10, no. 5, p. 056016, 2015. (Chapter 4)