



University of Sussex

**A University of Sussex DPhil thesis**

Available online via Sussex Research Online:

<http://sro.sussex.ac.uk/>

This thesis is protected by copyright which belongs to the author.

This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the Author

The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the Author

When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given

Please visit Sussex Research Online for more information and further details

# **Bio-Inspired Approaches to the Control and Modelling of an Anthropomorphic Robot**

**Alan Diamond**

**Submitted for the Degree of D.Phil.  
University Of Sussex  
June 2013**

**Declaration**

I hereby declare that this thesis has not been and will not be, submitted in whole or in part to another University for the award of any other degree.

Signature:.....

## Summary

Introducing robots into human environments requires them to handle settings designed specifically for human size and morphology, however, large, conventional humanoid robots with stiff, high powered joint actuators pose a significant danger to humans. By contrast, “anthropomimetic” robots mimic both human morphology and internal structure; skeleton, muscles, compliance and high redundancy. Although far safer, their resultant compliant structure presents a formidable challenge to conventional control. Here we review, and seek to address, characteristic control issues of this class of robot, whilst exploiting their biomimetic nature by drawing upon biological motor control research. We derive a novel learning controller for discovering effective reaching actions created through sustained activation of one or more muscle synergies, an approach which draws upon strong, recent evidence from animal and humans studies, but is almost unexplored to date in musculoskeletal robot literature.

Since the best synergies for a given robot will be unknown, we derive a deliberately simple reinforcement learning approach intended to allow their emergence, in particular those patterns which aid linearization of control. We also draw upon optimal control theories to encourage the emergence of smoother movement by incorporating signal dependent noise and trial repetition.

In addition, we argue the utility of developing a detailed dynamic model of a complete robot and present a stable, physics-based model, of the anthropomimetic ECCERobot, running in real time with 55 muscles and 88 degrees of freedom.

Using the model, we find that effective reaching actions can be learned which employ only two sequential motor co-activation patterns, each controlled by just a single common driving signal. Factor analysis shows the emergent muscle co-activations can be reconstructed to significant accuracy using weighted combinations of only 13 common fragments, labelled “candidate synergies”. Using these synergies as drivable units the same controller learns the same task both faster and better, however, other reaching tasks perform less well, proportional to dissimilarity; we therefore propose that modifications enabling emergence of a more generic set of synergies are required.

Finally, we propose a continuous controller for the robot, based on model predictive control, incorporating our model as a predictive component for state estimation, delay-compensation and planning, including merging of the robot and sensed environment into a single model. We test the delay compensation mechanism by controlling a second copy of the model acting as a proxy for the real robot, finding that performance is significantly improved if a precise degree of compensation is applied and show how rapidly an un-compensated controller fails as the model accuracy degrades.

## **Acknowledgements**

I would particularly like to thank and acknowledge my first supervisor Owen Holland for his unstinting advice, support effort and honesty. I would also like to thank my second supervisor Anil Seth for his belief in me, and advice during my first doctoral year. Finally, my wife Catherine, for sticking with me over these last few years and helping me to keep it real.

# Contents

---

<b>Chapter 1 Introduction</b> .....	<b>1</b>
1.1 Glossary of Terms .....	1
1.2 Thesis Overview .....	3
1.3 Summary of Original Contributions .....	9
1.4 List of publications arising from this work.....	10
1.4.1 First Author / Joint First Author Publications and Submissions .....	10
1.4.2 Publications submitted.....	11
1.4.3 Conference Abstracts / Posters .....	11
1.4.4 Contributing Author Publications.....	11
1.4.5 Published Software.....	11
<b>Chapter 2 : Background</b> .....	<b>12</b>
2.1 Overview .....	12
2.2 Musculoskeletal Humanoid Robots.....	13
2.2.1 The ECCERobot .....	13
2.2.2 Cronos.....	15
2.2.3 Other anthropomimetic and musculoskeletal humanoid robots .....	15
2.3 The Control Problem .....	16
2.4 Potential Control Approaches – An Overview.....	17
2.4.1 Classical Control .....	17
2.4.2 Motor planning search .....	18
2.4.3 Bio-inspired and learning approaches.....	19
2.4.4 Conclusion .....	24
2.5 Bio-inspired evidence underpinning control approach selection.....	25
2.5.1 Introduction.....	25
2.5.2 Standard Reinforcement Learning .....	25

2.5.3	Reinforcement learning for high dimensional state spaces and humanoid robotics.....	27
2.5.4	Morphological computation.....	28
2.5.5	Muscle synergies.....	30
2.5.6	Control approaches exploiting natural dynamics and compliance.....	35
2.6	Selection of Control Approach for ECCERobot .....	36
2.7	Control Target – robot or model? .....	38
2.8	Conclusions.....	39

**Chapter 3 Developing a physics-based model of a complete anthropomimetic torso under compliant muscle actuation.....41**

3.1	Overview .....	41
3.2	Capturing robot morphology to create a static 3D model .....	42
3.3	Selection of physics engine for dynamic simulation of robot .....	45
3.4	Preparing for simulation through analysis of joints, constraints and other issues 49	
3.5	Creating a physics-based model of the passive structure.....	50
3.5.1	Instability in the physics engine .....	50
3.5.2	Building the model incrementally .....	50
3.5.3	Implementation of specific modelling issues .....	51
3.6	Simulating the active structure .....	53
3.6.1	Approach for modelling of muscle cable forces.....	53
3.6.2	Strategy for adding muscles incrementally to the passive structure .....	54
3.6.3	Modelling bodies joined by elastic cable .....	54
3.6.4	Muscle cables that wrap bodies .....	54
3.6.5	Statistics of complete actuated model.....	56
3.7	Validating the model.....	56
3.7.1	Platform.....	56
3.7.2	Addressing spinal issues .....	57

3.7.3	Achieving a standing state .....	57
3.7.4	Testing responses to muscle control signals .....	58
3.8	Control research undertaken using the physics model.....	58
3.9	Conclusion .....	59
<b>Chapter 4 : Controlled Reaching Exploiting Motor Synergies Emergent Under Reinforcement Learning Part I: Algorithm Development .....</b>		<b>60</b>
4.1	Introduction.....	60
4.2	Overview .....	61
4.3	Problem State.....	65
4.4	Actions.....	66
4.4.1	Parameterising an action as a signal-driven muscle co-activation.....	66
4.5	Policy .....	68
4.5.1	Action value.....	69
4.5.2	Exploiting continuous state space to create noise driven exploration.....	69
4.5.3	Creating a new action from weighted combination of stored actions .....	70
4.6	Trial and Reward .....	70
4.6.1	Reward Function.....	70
4.6.2	Trial repetition and signal dependent noise to leverage optimal control ....	71
4.7	Policy update.....	72
4.7.1	Updating values of stored SAR contributors.....	72
4.7.2	Constructing and reappraising new action based on best problem state ....	73
4.7.3	Assigning reward to new actions.....	74
4.7.4	Limiting stored plans to encourage emergence of an effective dominant set	
	75	
4.8	Complete learning algorithm.....	75
4.9	Initial feasibility investigation .....	77
4.9.1	Creating functional movements using action definition and parameter set	77
4.9.2	Testing action combination algorithm.....	77

4.9.3	Tonic muscle activation .....	77
4.9.4	Trial repetition to exploit optimality theory .....	78
<b>Chapter 5 : Controlled Reaching Exploiting Motor Synergies Emergent Under Reinforcement Learning Part II: Experiments and Results .....</b>		<b>80</b>
5.1	Overview .....	80
5.2	Learning to perform a reaching task.....	81
5.2.1	Introduction.....	81
5.2.2	Problem State.....	81
5.2.3	Target Object and Placement .....	81
5.2.4	Definition of a reaching movement.....	82
5.2.5	Reward Function.....	84
5.2.6	Dimensionality of muscle space .....	86
5.2.7	Initial set of stored actions.....	86
5.2.8	Policy Function.....	87
5.2.9	Creating new SAR to store .....	88
5.3	Implementation and experiment parameters .....	89
5.3.1	Implementation platform .....	89
5.3.2	Stored SAR limit.....	89
5.3.3	Clocking rates for the simulator and physics model.....	89
5.3.4	Learning trial duration and repetitions .....	89
5.4	Results .....	90
5.4.1	Reaching to the target.....	90
5.4.2	Primary issues encountered .....	93
5.4.3	Looking for emerging signatures of optimality .....	94
5.4.4	Exploiting biomechanical structure .....	97
5.4.5	Emergent muscle activation patterns and synergies .....	102
5.5	A reaching controller based on fixed synergy units.....	113
5.5.1	Introduction.....	113

5.5.2	Method .....	113
5.5.3	Results .....	114
5.5.4	Learning different new tasks using the synergy-based controller .....	116
5.5.5	Composition of emergent synergies .....	119
5.6	Discussion and potential implications of the findings .....	121
5.6.1	Introduction .....	121
5.6.2	Transfer of approach to physical robot.....	121
5.6.3	Biological Implications.....	122
5.6.4	Limitations of the study .....	126
5.7	Future Work .....	129
5.7.1	Learning core synergies applicable across tasks .....	129
5.7.2	Incorporation into general control architecture based on MPC .....	130
5.7.3	Extending the problem space – commencing from any state.....	130
5.7.4	Trajectory storage approach for speeding of learning for physical robot..	132
5.7.5	Comparative studies with alternative methods.....	133
5.7.6	Creating hybrid, synergy-based control approaches .....	134
5.8	Conclusion .....	134
<b>Chapter 6 A bio-inspired continuous control architecture for an anthropomimetic robot incorporating environment integration and delay-compensation .....</b>		<b>137</b>
6.1	Introduction.....	137
6.2	Issues Arising In the Design of Robot Controllers.....	139
6.3	Principles of model predictive control (MPC).....	140
6.4	Use of the ECCERobot Physics-Based Model.....	141
6.5	Proposed Design .....	141
6.5.1	Addressing sensor noise and inaccuracy .....	141
6.5.2	Planning .....	143
6.5.3	Environment Capture .....	143
6.6	Delays .....	144

6.6.1	Effects of sensorimotor delay .....	144
6.6.2	Causes of delay .....	145
6.6.3	Combating Delay .....	145
6.6.4	Predicting intended motor signals with an updateable “buffer” .....	145
6.6.5	Delay compensating design .....	146
6.7	Experiment exploring delay compensation .....	148
6.7.1	Overview .....	148
6.7.2	Method .....	149
6.7.3	Characterising Effects of Model Divergence .....	154
6.8	Discussion and Conclusion .....	156
<b>Chapter 7 : Conclusion .....</b>		<b>159</b>
7.1	Aims of the thesis.....	159
7.2	Original Contributions of Thesis .....	160
7.2.1	A physics-based forward model of a complete musculoskeletal robot torso 160	
7.2.2	Simple reinforcement learning can produce reaching control of complex, musculoskeletal robot model by using an approach of muscle co-activations, simple shared driving signals and natural dynamics.....	161
7.2.3	A low dimensional reaching controller for biomimetic musculoskeletal modelled robot based on extracted emergent synergies .....	162
7.2.4	Optimal control principles can be exploited through RL trial repetition to refine movements.....	162
7.2.5	Biological implications: support for synergy-based motor control theories 163	
7.2.6	An MPC-based design for continuous control of an anthropomimetic robot incorporating delay compensation.....	163
7.3	Future Work .....	164
<b>Chapter 8 : References .....</b>		<b>165</b>
<b>Chapter 9 : Appendices .....</b>		<b>179</b>

9.1	Appendix I: Physics engine comparison report .....	180
9.1.1	PhysX / Novodex .....	181
9.1.2	BULLET .....	185
9.1.3	ODE.....	191
9.1.4	Havok .....	195
9.2	Appendix II: ECCERobot Project Report on Model Construction .....	199

## List Of Figures

---

Figure 1. Humanoid and anthropomimetic robots – Asimo, Cronos, ECCERobot.....	3
Figure 2. Biomechanical construction details of ECCERobot.....	13
Figure 3. Compliant muscle-based motor actuation design used in the ECCERobot ....	14
Figure 4: Using an intermediate layer to linearize control.....	29
Figure 5: Model of muscle pattern generation by a combination of muscle synergies (reproduced from Cheung et al, 2009) .....	32
Figure 6. Selection of source material generated for reverse-engineering of ECCERobot model .....	43
Figure 7. Stages in development of static model using the Blender tool.....	44
Figure 8. Reverse-engineered static model of ECCERobot torso (Front and Side) .....	46
Figure 9. Textured rendering of static model of ECCERobot using translucent bones to show interior.....	48
Figure 10. Screenshot of spreadsheet compiled detailing every constraint and proposed implementation .....	49
Figure 11. Stages in the migration of the static Blender model to a definition of the physics model in the Bullet engine.....	51
Figure 12. Defining a physics based model as an XML file .....	51
Figure 13. Physics-based model: the floating shoulder blade and arm .....	52
Figure 14. Design of virtual spherical pulleys for muscle wrapping .....	55
Figure 15. Stable standing under tensioned muscles.....	58
Figure 16. Standard Reinforcement Learning Cycle .....	62
Figure 17: Action generation, trial and storage .....	64
Figure 18: Anatomy of an Action; Driving a co-activation pattern of muscle motors to cause a movement by the body .....	67
Figure 19: Parameterised driving signal used to control waveform of the motor input voltage signal.....	67
Figure 20: The final learning flow algorithm .....	76
Figure 21: Side and top view of reaching experiment .....	82

Figure 22: Anatomy of a compound action.....	83
Figure 23. Examples of successful reaching to the target.....	90
Figure 24. Distribution of trial outcomes at six stages of learning.....	91
Figure 25. Average reward by type issued per trial over 1000 trials.....	92
Figure 26. Specific model and control issues encountered in learning .....	94
Figure 27. Signatures of optimality - reaching profiles and reliability change .....	96
Figure 28. Optimality under noise; changes in reaching reliability and conformance to bell curve stereotype during learning .....	98
Figure 29. Exploiting the biomimetic aspects of the of structure; compliance, full-body dynamics and biomechanical structure .....	100
Figure 30. Distribution of target locations linked to stored motor plans.....	102
Figure 31. Eight reaching zones defined for muscle co-activation pattern analysis .....	102
Figure 32. Parameterised driving signals of most valued reaching actions with targets spread across eight sub-zones .....	104
Figure 33. Muscle-coactivation patterns of most valued reaching actions for a target in each of eight sub-zones .....	105
Figure 34. Effect on learned reaching performance of continually reducing set of stored actions .....	106
Figure 35. Distribution of target locations with a minimised set of stored actions (40 entries) .....	107
Figure 36. Co-activation data accounted for by weighted combinations of candidate synergies uncovered using factor analysis.....	110
Figure 37. Thirteen muscle synergies extracted by factor analysis of muscle co-activation patterns.....	111
Figure 38. Reconstruction of co-activation patterns from 13 extracted candidate synergies.....	112
Figure 39. Synergy-based controller - average reward by type issued per trial over 1000 target presentations.....	115
Figure 40. Synergy-based controller learning the original problem (a) and extended problems (b,c) .....	115
Figure 41. Performance of three reaching tasks learned by synergy-based controller .....	117

Figure 42. MPC-based robot controller using a physics engine to capture and predict dynamic state of robot and environment .....	142
Figure 43. Motor signal buffer design for an MPC-based controller for the ECCERobot .....	146
Figure 44. Schematic of full MPC-based controller for the ECCERobot incorporating delay compensation mechanism.....	147
Figure 45. The impact of sensorimotor delay on planning and use of compensation ..	149
Figure 46. Experimental configuration to characterise the performance of the delay compensation design .....	150
Figure 47. Delay-compensated reaching performance over a range of fixed system delays and reaching commencement positions .....	153
Figure 48. Effects of model accuracy on delay-compensated reaching performance of simulated ECCERobot reaching.....	155

## List Of Tables

---

Table 1. Stabilising statistical endpoint variation.....	78
Table 2. Detail and discussion of synergy composition .....	120
Table 3. Extending dimensions of the problem state .....	131

# Chapter 1

## Introduction

---

### 1.1 Glossary of Terms

In this thesis we refer to a number of common terms which are used in the bio-mechanical and musculoskeletal robot literature but can imply a variety of meanings. For the avoidance of doubt we therefore first define here the most important, using the terms in which we will be using them in this thesis.

<i>Muscle</i>	In addition to its biological meaning, in the context of controlling the biomimetic robot in question, a <i>muscle</i> implies the actuator formed by the aggregation of an electric motor, pulley, inextensible winding cable, elastic cord and attachment points to the “bone”, simulating the role of a true compliant muscle in a biological body.
<i>Muscle activation</i>	Implies applying a voltage-based <i>driving signal</i> to a motor, causing the <i>muscle</i> to be correspondingly activated, generating an actuation force between the bodies to which it is attached.
<i>Driving signal</i>	A constant or varying signal applied as a voltage waveform to drive the activation of one (or more) muscles.
<i>Muscle co-activation</i>	This refers generally to any simultaneous <i>activation</i> of two or more <i>muscles</i> . However, the muscles in question might be driven by individual driving signals or they may share a single driving signal.
<i>Co-activation pattern</i>	In the case where a driving signal is shared, each activated muscle may respond in different proportions to the signal. This proportional response can be illustrated as a pattern of relative weightings across the set of muscles involved.

- Synergy* A fixed *co-activation pattern* spanning a subset of muscles sharing a single driving signal. Strong evidence suggests that superimposing a simple combination of synergies together is a simple way to form a complex resultant overall muscle co-activation that can perform effective motor tasks.
- Candidate synergy* We use “candidate” to refer to a synergy pattern identified by analysis (generally a variant of component or factor analysis) of muscle activation data as recurring across multiple (often differing) movements. Numerous biological studies have shown that activation data can very often be largely reconstructed by a simple superposition of only a few candidate synergies. This post-hoc identification empirically suggests, but does not prove, that the synergy patterns are being specifically employed as distinct units by the controller to generate movement.
- True synergies* We use this to refer to a set of synergy patterns that have been explicitly used to generate the muscle co-activations that result in movements that are effective in solving a presented task, such as reaching to a target.
- Hierarchical synergy* This refers to the concept of a “synergy-of-synergies” where the co-activation formed by the superposition of two or more synergies might itself be treated as a synergy-like unit at a higher level. This might conceivably generate, for example, coordinated movements across disparate body parts, such as a swing forward of one arm, with the swing back of another.

## 1.2 Thesis Overview

The useful and effective introduction of robots into human environments requires them to manage settings and scenarios designed specifically for human size and morphology, such as stairs or door handles. However, conventional humanoid robots, such as the well known Asimo (Figure 1a), are equipped for precision control via high powered, stiff joint actuators and would constitute, at life-size, a significant danger to humans. Their engineering also makes them of little interest with regard to providing insight into human motor control. By contrast, so-called “anthropomimetic” robots (Holland & Knight 2006) such as ECCERobot (Wittmeier et al. 2013; Marques et al. 2010) (Figure 1c), or its predecessor, Cronos (Holland et al. 2010) (Figure 1b), attempt to replicate not just morphology, but the internal “musculoskeletal” structure; bones, muscles, tendons, complex joints and compliance. It is particularly this structural compliance that makes this class of robots potentially far safer than their conventional counterparts, however, their highly non-linear biomechanical structure presents a formidable challenge to conventional control methods. Nevertheless, their biomimetic nature also provides a clear opportunity for reciprocal research - we may study biological evidence with a view to uncovering effective control approaches, whilst conversely, functional controllers developed for these robots may make predictions that can be tested in biology. These may in turn provide insights into human motor control and even cognition.

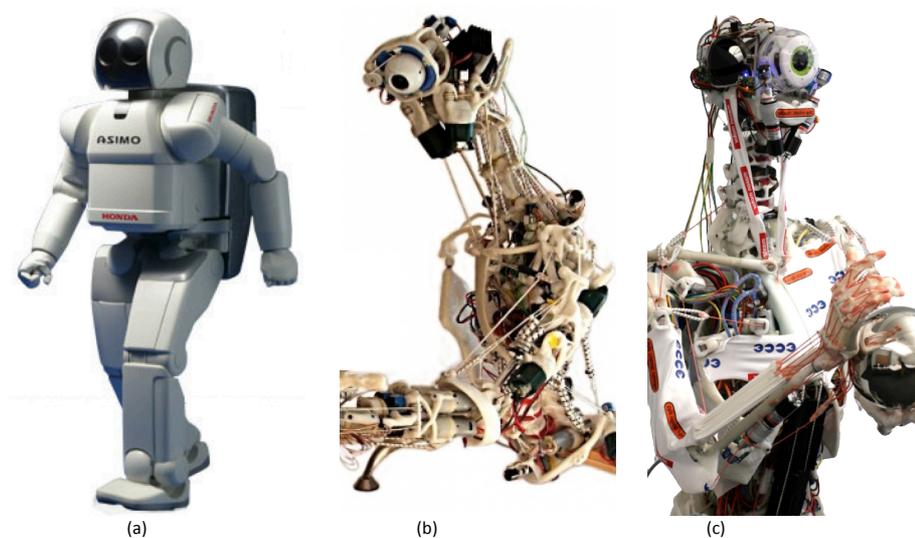


Figure 1. Humanoid and anthropomimetic robots – Asimo, Cronos, ECCERobot

This thesis is concerned with developing methods that address the characteristic control issues that arise with this class of robot whilst exploiting their biomimetic nature, both to draw upon biological motor control research and to potentially inform our knowledge or theories of biological motor systems. The contents are structured into five main chapters as follows:-

In *Chapter 2* we review the anthropomimetic *ECCERobot* and its musculoskeletal peers and discuss control methods of interest developed to date for this new class of robots, concluding that very few of note exist as yet. We therefore take a step back to consider the particular issues of the control problem and review at a high level a range of potential control approaches, including evidence from biological motor systems. We conclude that bio-inspired approaches hold the most promise for controlling such biomimetic structures, structures that would be considered highly complex, high dimensional control subjects by more conventional engineering control approaches or planning search approaches.

We therefore review in greater detail a range of bio-inspired approaches, with a view to selecting for investigation one with a strong combination of novelty, promise, and interest. In particular, we focus on recent strong evidence from biological studies suggesting that- in contrast to conventional theories - effective control of seemingly highly complex structures, such as the bodies of frogs, cats or humans, is achieved largely through advantageous, co-evolved natural dynamics (morphological computation) combined with a relatively low number of simple, shared activation signals each driving a number of fixed, yet precisely weighted precise muscle groupings (synergies). We therefore derive from the review the following primary research question.

***Primary Research Question*** - A promising and relatively novel study would test the hypothesis, arising from strong biological evidence, that applying a muscle group co-activation approach to an extensive, yet biomimetic structure with potentially rich natural dynamics - such as the *ECCERobot* - may allow significantly simpler learning techniques to be deployed than the complex algorithms under development for generic high dimensional control subjects in fields such as reinforcement learning, force control or planning search.

Of these simpler methods, we choose to trial an approach built primarily upon reinforcement learning fundamentals, citing as reasons its bio-inspired nature and its “action discovery” potential for exploiting natural dynamics of the full body, over conventional precise trajectory plotting.

We also argue that for exploring, at an early stage, potential avenues for effective control a detailed dynamic model closely approximating the complete robot, would prove of great benefit. This would provide a fast, convenient and realistic platform for trialling control approaches or for extended periods of offline learning or planning search. Secondly, such a model may also potentially form an important component in a predictive internal-model based controller architecture, offering features such as delay compensation and Kalman-filter optimised state estimation. Nevertheless, the overarching goal of achieving control of the real robot remains important and potential transference of an approach from the model to the physical robot is given consideration whenever possible.

To construct such a model we therefore briefly review the available full body models and musculoskeletal model building tools, concluding that none offer the necessary direct muscle activation input nor the ability to model the robot environment – a crucial element for motor planning in the real world. We therefore propose the use of a fast, modern physics simulation engine for the construction of a novel, detailed physics-based model of a complete anthropomorphic robot, incorporating the potential to exploit full body natural dynamics and interaction with sensed environment objects, which themselves may be modelled dynamically.

In *Chapter 3* we detail our design and engineering of a complete physics-engine based model robot, reverse-engineered from one of the anthropomorphic ECCERobot prototypes. Custom-modelled components include the elastic muscles, motors, gearboxes, pulleys and joint friction. A stable model is presented running in real time with 55 muscles and 88 degrees of freedom that can act as a subject of near-equivalent complexity to the robot for our primary investigation into the control of such structures.

In *Chapter 4* we present a design for a novel learning controller for discovering effective reaching actions driven by the sustained, weighted activation of a set of muscle co-activation patterns, drawing upon on the evidence from muscle group synergy research in frogs and humans. As the most effective muscle activation patterns and driving signals for the ECCERobot are unknown, we test a simple reinforcement-learning based approach intended to allow effective muscle groupings to emerge. By allowing only a simple driving signal shared in *linear* weighted proportion amongst muscle groupings we seek specifically to encourage the emergence of those activation patterns that act to linearize the control of the underlying non-linear structure. In addition, we draw upon optimal control theories to encourage the emergence of smoother, more natural movement, by incorporating signal dependent noise and trial repetition into the learning cycle.

In *Chapter 5* we present experiments testing this approach in learning control of the modelled ECCERobot to perform reward-based reaching tasks, aiming to touch or strike a series of randomly placed target objects. Notably, we find that, far from requiring the accurate trajectory control, individual motor signals and precise high speed sensing of conventional control, we find that reaching actions can be generated surprisingly successfully employing only two sequential sets of motor co-activation patterns. It is notable that each set of co-activations is simply driven, in weighted proportion, by a single shared motor activation. We suggest that the resultant very large reduction in the dimensionality of the search space encourages a purely reward-driven “action discovery” approach to succeed by drawing heavily on amenable natural dynamics of the biomimetic structure.

Furthermore, applying factor analysis techniques to the muscle activation signals generated from trials shows that the set of activation patterns emergent during the learning can be reconstructed to 80% accuracy using only weighted combination of 13 common fragments. We label these emergent fragments *candidate synergies*, since we define a “true” synergy as a pattern that is used explicitly up front, driven as a single unit, by the controller to generate motor signals.

To test if these candidates can act as true synergies we therefore test a reworked controller design that drives, not individual muscles, but the identified set of

candidate synergies. We find that this controller learns the same task both faster and with better performance, however, other related (but different) reaching tasks perform proportionally less well. We therefore judge that, although a method alteration or extension is required to identify a more generic (i.e. widely applicable) set of core synergies, the candidate synergies that were located via analysis can indeed be effectively employed as a valid set of true emergent synergies and we examine their constituent parts to analyse whether they emerge with specific roles in generated reaching movements.

Reinforcement learning has a potential as an action discovery mechanism, i.e. uncovering solutions that fulfil the task through exploration rather than a prescriptive, calculated trajectory. We therefore also present evidence that the learning has employed amenable natural dynamics of the biomimetic structure to generate solutions to reaching tasks.

In unreliable or noisy systems, reinforcement learning will, over repetitions, inherently favour the most reliable solutions as they will accrue the most reward (Wolpert et al. 2001). Using optimal control theory, Harris and Wolpert (1998) have shown that smooth movements observed in nature, such as when reaching to grasp, (as typified by “bell-curve” velocity profiles) can be explained by a cost function that minimises endpoint variance (i.e. maximises reliability) when in the presence of amplitude-related motor neuron noise (Harris & Wolpert 1998). In other words, the movements selected are those which are most reliable over repetition, a very clear benefit when subjected to this form of observed neural noise. We therefore test whether we can similarly encourage the emergence of smoother movement by incorporating both signal dependent noise and trial repetition into the learning process. We find that for those regions where the controller has learned to significantly slow the robot’s hand for arrival at the target, we do observe over a period of learning a migration towards the stereotype bell-curve “signature of optimality” velocity profile. Across all targets regions we also observe an increasing smoothness of movement (reduction in jerk) and an increase in reliability. Furthermore, as predicted by Harris and Wolpert (1998), these results applied when adding signal-dependent Gaussian noise, but not for fixed-level Gaussian noise.

Compliance is a primary feature that sets both biological bodies and these musculoskeletal robots apart from conventional stiff-jointed robots. This elasticity is

one of the key features that can potentially offer significant greater safety to humans in proximity to a large robot but can add significantly to the complexity of conventional control approaches. We therefore conduct some preliminary comparison trials to inform on the effects of compliance in aiding or hindering our approach in its control of complex musculoskeletal structures. Initial results suggest that the compliance in our model contributes to a reduction in jerk, thereby smoothing movement, and furthermore, acting as an energy store allowing for a reduction in the motor force needed for direction changes, resulting in a drop in signal related noise that causes unreliability. We discuss some potential implications for both robot design and insights into biological motor control.

Lastly, in *Chapter 6*, we discuss how to implement a continuous controller for such a robot and in particular the issues introduced by sensorimotor delays when dealing with a highly dynamic and compliant structure. We propose a delay-compensating continuous controller design based on the principles of *model predictive control* which draws upon our physics-based model as a predictive component for state estimation, delay-compensation and planning. It also includes employing the physics engine as an integrated simulation container for merging of the model and sensed environment. We demonstrate its effects on controlling a second copy of the model acting as a proxy for the real robot, showing that performance is significantly improved if a precise degree of delay compensation is applied. Furthermore, we show, by a controlled degradation of our model's accuracy, that as the model dynamics diverges from that of the "robot" under control, a controller without compensation rapidly performs very poorly. Finally, we discuss possible implications and questions around human cognition and perception of "the present".

The final *Chapter 7* reviews the research, puts forward a case for its original contributions and draws overarching conclusions from the full thesis.

### 1.3 Summary of Original Contributions

Here we summarise the original contributions asserted in this thesis and in resultant publications.

- We present a complete physics-engine based simulation model of a musculoskeletal robot, reverse-engineered from a real anthropomorphic robot constructed using Grays Anatomy as a guide (Diamond & Holland 2012). The dynamic model runs in real time and incorporates simulations of the muscles, motors, gearboxes, pulleys and joint friction (Wittmeier et al. 2011). A stable version is available with 55 elastic muscles and 88 degrees of freedom that can act as a biomimetic structure of high complexity.
- We present a design for a novel learning controller for a complex full-body musculoskeletal, compliant structure employing a combination of bio-inspired approaches; namely, muscle synergies, reinforcement learning and natural dynamics.
- We demonstrate the design as effective in learning muscle activation patterns that control a complex physics modelled simulation of a complete anthropomorphic robot to produce reaching to sequentially presented, randomly positioned targets.
- Using factor analysis of 100 emergent muscle co-activation patterns we demonstrate 13 distinct emergent fixed-weighting “candidate” synergies that can reconstruct the original set in simple weighted combination. We demonstrate that a faster learning and higher performing controller can be created by driving weighted combinations of the emergent synergies instead of individual muscles.
- An additional contribution of the study is experimental support for the use of reward issued in repeated trials to bring about increased endpoint reliability under signal-dependent Gaussian noise, resulting in smoother and increasingly naturalistic movement in a biomimetic structure - as judged by chi-squared similarity to the well known bell-curve velocity profile observed in nature.
- The studies also contribute informed opinion on the transferability of this model-tested approach to the control of the real robot.
- We derive a control architecture for the real robot, based on the proven Model Predictive Control, incorporating the physics model as a predictive component

for proprioception correction, delay-compensation and planning, including the merging in the physics simulation of the robot and sensed dynamic and static elements from its environment.

- We demonstrate the effect of the delay compensation mechanism by controlling a second copy of the model acting as a proxy for the real robot, showing that performance is significantly improved if a precise degree of delay compensation is applied. Finally, we show by a controlled degradation of our model that as the model dynamics diverges from that of the “robot” under control, a controller without compensation rapidly performs very poorly.

## **1.4 List of publications arising from this work**

### **1.4.1 First Author / Joint First Author Publications and Submissions**

Wittmeier, S., Diamond, A. et al., 2013. Toward anthropomimetic robotics: development, simulation, and control of a musculoskeletal torso. *Artificial life*, 19(1), pp.171–93.

My contribution to this journal paper was the chapter presenting initial reaching experiment results for the learning controller described in this thesis (See Chapters 4 and 5).

Diamond, A. et al. 2012. Anthropomimetic Robots: Concept, Construction and Modelling. *International Journal Of Advanced Robotic Systems*. My contribution to this journal paper was the section (around 50% of the total) covering the process of building a physics model of the ECCERobot.

Diamond, A. , Holland,O., & Marques, H. 2011. *The role of the predicted present in artificial and natural cognitive systems*. Proceedings of the Second Annual Meeting of the BICA Society.

My contribution to this conference paper was the section (around 50% of the total) that introduces a design for a delay compensating predictive controller using a physics-based model. This design, and testing of the delay compensation, are covered in much greater detail in Chapter 6 of this thesis.

### 1.4.2 Publications submitted

A.Diamond & O.Holland. 2013. *Reaching control of a full-torso, modeled musculoskeletal robot using muscle synergies emergent under reinforcement learning*. *Bioinspiration & Biomimetics*, IOP Journal. Abstract accepted. Full paper in peer review as of June 2013.

This journal paper focuses on fully detailing the muscle co-activation and synergy-based control learning approach, including both the learning algorithm (see Chapter 4) and experiments (see Chapter 5).

### 1.4.3 Conference Abstracts / Posters

Diamond, A. & Holland, O. 2012. No time like the present? Potential anomalies in time perception exposed by anthropomimetic robot control research. *Association for Scientific Study of Consciousness, ASSC16 Conference, July 2012*, Poster presentation.

Diamond, A. et al. GPU-Powered Control of a Compliant Humanoid Robot. *GPGPU Technology Conference, Oct 2010*, Poster presentation.

### 1.4.4 Contributing Author Publications

Holland,O., Diamond, et al. 2012. Real and apparent biological inspiration in cognitive architectures. *Journal of Biologically Inspired Cognitive Architectures (BICA)*.

Holland, O., Diamond, A., Mitra, B., & Devereux, D. 2011. The What, Why and How of the BI in BICA. *Proc. of the Second Annual Meeting of BICA Society*, pp138-145.

Devereux, D., Diamond, A. et al. 2011. Using the Microsoft Kinect to model the environment of an anthropomimetic robot. *Proc. of the 2nd IASTED Intl. Conf. on Robotics (Robo2011)*.

Marques, H., Diamond, A. et al., 2010. ECCE1: the first of a series of anthropomimetic musculoskeletal upper torsos. *In 10th IEEEERAS International Conference on Humanoid Robots*. IEEE, pp. 391–396.

### 1.4.5 Published Software

The full output of the ECCERobot project including documentation is available via [www.eccerobot.org](http://www.eccerobot.org) and the software, including the physics based model of the full robot contributed by this thesis (see Chapter 3) are available under open-source licence.

## Chapter 2 : Background

---

### 2.1 Overview

In this chapter we review the anthropomorphic *ECCERobot* and other musculoskeletal robots and discuss control methods developed to date for this new class of robots.

We discuss the particular issues of the associated control problem and consider the suitability of a number of established or emerging control approaches, including evidence from biological motor systems. We conclude that bio-inspired approaches hold the most promise for controlling a biomimetic structure that would be considered highly challenging by conventional robot controllers.

We therefore review in greater detail a range of bio-inspired approaches with a view to selecting for investigation one with a strong combination of novelty, promise, and interest. In particular, in contrast to prevailing theories, we focus on recent strong evidence from biological studies demonstrating the extent to which effective motor control of frogs, cats or humans draws heavily upon a combination of advantageous, co-evolved natural dynamics and simple fixed-weight activations of precise muscle groupings (synergies).

We conclude from the evidence that a promising and relatively novel study would test the hypothesis that drawing upon a muscle group co-activation approach for an extensive biomimetic robot structure with potentially rich natural dynamics may facilitate significantly simpler search and learning techniques to be deployed than the complex algorithms currently under development for generic, high-dimensional control subjects. Of these simpler methods, we choose to trial an approach built primarily from reinforcement learning (RL) fundamentals, citing as reasons the bio-inspired nature and “action discovery” potential of RL for exploiting natural dynamics of the full body.

Finally, we consider whether our selected approach should be developed against the physical robot or a modelled approximation, at least for preliminary investigations.

We briefly review available full body models and musculoskeletal model building tools, concluding that none are fit for the purpose of an anthropomorphic robot controller. We therefore propose employing a fast, modern physics simulation engine to construct a complete physics-based model which incorporates actuation modelling, demonstrates full body natural dynamics and can potentially predict dynamic interaction (e.g. collision) with sensed environment objects.

## 2.2 Musculoskeletal Humanoid Robots

We review the anthropomorphic *ECCERobot* and other musculoskeletal robots and discuss control methods developed to date for this new class of robots.

### 2.2.1 The ECCERobot

#### 2.2.1.1 Introduction

The ECCERobot is the latest in a line of so-called “anthropomorphic” robots that began with the robot Cronos (Holland & Knight 2006), and which are human-sized, human-

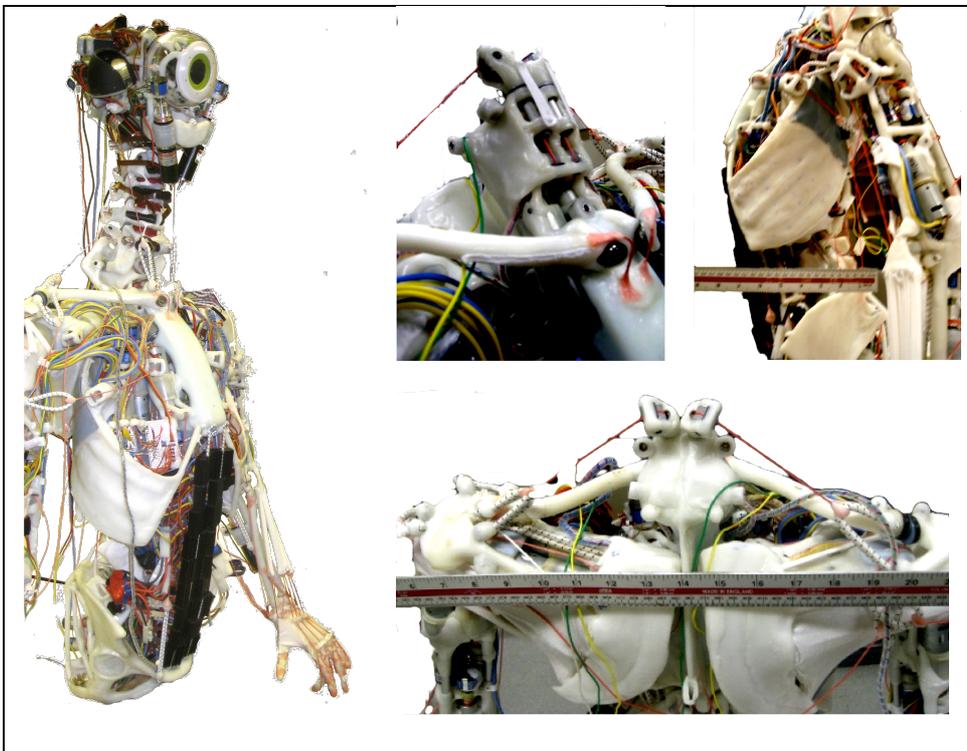


Figure 2. Biomechanical construction details of ECCERobot

shaped, and have human-like biomechanical construction. A distinguishing feature of these robots is that, as well as compliant actuation, they also look to mimic the skeleton, joints and muscle attachment points. Each iteration of the ECCERobot has looked to extend its biomimetic nature, using Grays Anatomy (Gray 1901) as a direct guide to construction (Wittmeier et al. 2012; Holland et al. 2010).

### 2.2.1.2 *Skeleton and joints*

The biomechanical structure of the ECCERobot is illustrated in Figure 2. The robot torso has a skeleton of “bones”, hand-moulded from the low melting-temperature polymer polycaprolactone, commonly known as “polymorph”. In the majority of cases, these bones are connected with flexible joints with up to 6 degrees of freedom (DOF) often using kiteline or shockcord cabling to imitate ligaments, although some few, such as the elbow, are more precise 1DOF hinges. The total degrees of freedom approaches one hundred. The construction follows Grays Anatomy (Gray 1901) and includes floating shoulder blades that hang from the clavicles (collar bones) and dislocateable ball joints in the shoulders. The robot has a flexible spine with individual vertebrae and deformable foam discs, meaning that, just as for a human, it cannot stay upright without tensed muscles.

### 2.2.1.3 *Muscles and motors*

Figure 3 illustrates the compliant actuation used in the ECCERobot. The 50+ “muscles” of the ECCERobot are implemented as cables formed from a length of thin inelastic “kiteline” and attached to the bones via sections of elastic “bungee” shockcord that provide the compliance. A muscle is tensed through shortening the cable by winding

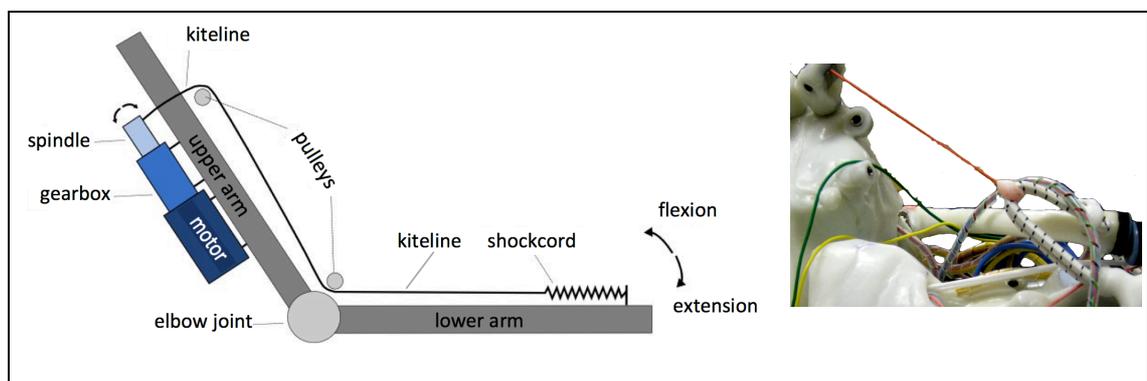


Figure 3. Compliant muscle-based motor actuation design used in the ECCERobot

onto the spindle of an individually-assigned high torque DC motor. Both high power, precision Maxon motors and low-cost electric screwdriver motors have been employed in this role. These motors are mounted on the skeleton and the cables routed, where necessary, by a series of pulleys. Muscles are relaxed simply by unwinding the cable by reversing the motors. However, there is no ability to vary the muscle stiffness or compliance as humans are known to do. The coefficient of elastic of shockcord is approximately constant up to the limit of extension, making the cable tension linearly proportional to the extension within the working range.

#### **2.2.1.4 Sensors**

The initial version of the robot included a webcam-based vision system, this was replaced later by a head-mounted Microsoft Kinect (Microsoft 2013) sensor enabling real-time 3D environment capture in the form of point clouds. These are processed to generate faceted surfaces and planes for live insertion into the Bullet Physics engine (Coumans n.d.) for use in motor planning tasks (see Devereux et al. 2011).

The robot also includes relatively minimal proprioception in the form of tension sensors placed in series with the muscle cables. Muscle length sensing is also available from motor-mounted encoders.

#### **2.2.2 Cronos**

The Cronos project (Holland & Knight 2006; Holland et al. 2010) preceded the ECCERobot, essentially forming a construction prototype for the later work, being also truly “anthropomimetic” in design. Only rudimentary control was ever established over the structure using a very limited set of muscles.

#### **2.2.3 Other anthropomimetic and musculoskeletal humanoid robots**

Apart from the ECCERobot and Cronos, there are no other extensive body robots at present that attempt to so closely mimic human construction in detail. However there are a growing number that employ musculoskeletal elements, primarily muscle-like compliant actuation. Examples of this class are “Kojiro” (Mizuuchi et al. 2007) and “Lucy” (Vanderborght et al. 2004). These remain in essence conventionally constructed robots albeit with some compliant actuation. With the focus strongly on the physical engineering challenges, control research with these robots has been limited to date, comprising primarily classical based approaches and biomimetic and bio-inspired approaches remain essentially unexplored. More recently, a number of robotic “passive

walkers” with compliant actuation have been constructed focusing specifically on the development of walking abilities. Examples of these are “Mabel” (Sreenath et al. 2009) and “BioBiped” (Scholz et al. 2011; Radkha & von Stryk 2012). However, these are also conventional robots with added compliant actuation and actively seek to be mathematically tractable to facilitate classical control approaches.

### **2.3 The Control Problem**

The ECCERobot presents an intrinsically challenging control problem with a very high dimensional state space and significant non-linearity. It has over 100 degrees of freedom, flexible joints with up to 6 DOF and complex structures such as floating shoulder blades and a flexible spine. It also has relatively very poor proprioception for its complexity, making accurate state capture impossible. Although there is significant friction in many joints this is not by design and the robot is essentially largely underdamped.

In fact, it can be argued that the ECCERobot presents a more formidable control subject than a human body itself which offers damping against oscillation, good proprioception, ultra-low friction fluid-encased joints and variable stiffness muscles. It also benefits from fine-tuned optimisation of structure and materials through evolution and goes through an extended period of development and growth (epigenetic staging) from the foetus stage upwards allowing the CNS to acquire control gradually (Lungarella et al. 2003; Bongard 2011). The full ECCERobot, by contrast, is presented as-is to any prospective controller.

Furthermore, the robot’s kinodynamic state (kinematic plus first derivative) can change rapidly with a highly non-linear response. As a result, any delays between sensing and acting will cause a correspondingly large issue for any controller. Of course, this is largely true for humans also, yet the brain, modified and informed by learning, appears to have solved these problems. This issue is potentially critical to control of this robot, therefore in *Chapter 6* we draw upon evidence from neuroscience to drive an investigation into the extension of controllers with predictive modelling and delay compensation components and test their effect on a physics-based model of the ECCERobot.

## 2.4 Potential Control Approaches – An Overview

In this section we present an overview of potential control approaches for this biomimetic subject, including evidence from biological motor systems. By doing so, we seek to identify a control approach with sufficient promise and interest to justify review and investigation in greater depth.

### 2.4.1 Classical Control

So-called “classical” methods, have been used extensively over a number of years as engineering solutions to robot control, where stability and resistance to perturbation are generally achieved by adding closed-loop feedback of one or more output variables (Franklin et al. 2002; Levine 1996; Sontag 1998). At their core, these methods seek to constrain the state space trajectory of the system to a satisfactory goal state by following a pre-calculated path generated using a model where the inputs (e.g. motor torques) and outputs are related by a set of differential equations. A *transfer function*, directly mapping input to output can be readily obtained if these equations are linear, but for increasingly non-linear systems correspondingly complex techniques must be brought to bear to calculate – or estimate – a transfer function (Atherton 2006; Sontag 1998).

In general therefore, these methods require a sufficiently tractable mathematical model of the subject and precise high-frequency state capture, their use is consequently largely limited to robots fulfilling such requirements, favouring the production of low-redundancy, stiff-jointed robots driven by high powered joint actuators mounted with precision sensors for state capture. In other words, very different from the class of robots we are dealing with here.

For this highly non-linear, high redundancy robot structure, to even describe its dynamics in the form required for this analysis has been proven highly challenging (Potkonjak et al. 2010) and may prove better suited, we would argue, to modelling in the kind of step-by-step approximation afforded by constraint-solving physics engines. Here, on each time step, the solver iterates the new state estimate towards one which better satisfies that set of constraints (e.g. joints) and forces (e.g. motor torques) describing the system at that point. The setting for the number of solver iterations is generally selected as a trade-off between performance and accuracy dependent on the application.

Nevertheless, a separate part of the ECCERobot project has been to ascertain the limits of classical methods applied to this class of robots. Investigations concluded that, although the trajectory control of a bi-articular model arm with two compliant muscles was achievable using a puller-follower design (Potkonjak et al. 2010) a more comprehensive, complex model with features such as floating shoulder blades could not be effectively controlled (Potkonjak et al. 2010).

#### **2.4.2 Motor planning search**

In contrast to classical control, planning search (Choset et al. 2005; Latombe 1991; LaValle 2006) is completely agnostic of the structure to be controlled and makes no assumptions about the likely form a motor plan might take. It is essentially an exercise in applying a series of test motor signals to a forward model of the system, with a view to rapidly exploring potentially high-dimensional state spaces as widely and efficiently as possible in a search for a route to a goal state. The rapidly-explored random tree (RRT) approach is one of the best known of these (LaValle & Kuffner 2001). Here, a branching “tree” of known valid paths through state space is built up by locating a new incremental movement from the nearest point on the current tree towards a random sample point in state space. Crucially, the combination of random new point with nearest known point causes the exploration to always branch, on average, towards the most unexplored regions, causing a rapid and even coverage of the space to be generated. This approach has been proven to be effective in even in the larger *kinodynamic* space (LaValle & Kuffner 2001). Kinodynamic implies a doubling of the dimensionality by extending the state vector beyond the kinematic state by adding the first derivative of the kinematic member variables. The core RRT technique has since been extended and accelerated. Some examples are: adding the ability to search for an optimum solution based on a cost function (Urmson & Simmons 2003); adding a macro search at low resolution (Sucan & Kavraki 2008); focusing the search on “useful” areas by attaching an updateable metric to each tree node (Burns & Brock, 2007); reducing the dimensionality by considering primarily those dimensions related to the goal task (Shkolnik & Tedrake 2009). It has also been applied with some success to (conventionally engineered) humanoid robotics (Kavraki et al. 1996; Kavraki 2007; Rusu et al. 2009; Ladd & Kavraki 2004; Kagami et al. 2003).

However, although the approach addresses the issue of high state space, this technique has core requirements that cannot be easily fulfilled for anthropomimetic robots such

as the ECCERobot. Firstly, it must be possible to generate truly random, valid state samples and to rapidly compare the proximity of two states. Secondly, a reliable rapid means to generate a motor signal that will move a known state towards a new one is needed, this requirement alone constitutes the construction of an inverse model of the robot. Thirdly, the motor plans generated are open-loop and must maintain the robot in a sufficiently stable state for continual use. Finally, a very fast forward model is required to cover the amount of exploration required to be effective in a very large state space. All of these requirements are significant issues without possessing a mathematically tractable model, which evidence from the classical control investigation suggests is unrealistic (Potkonjak et al. 2010). Sucas & Kavraki (2008) have estimated that sampling kinodynamic planners can spend up to 90% of their time in running forward propagation and sampling states. Nevertheless, it is conceivable that, in the future, a physics-based model “turbo-powered” by state of the art GPU-based acceleration may suffice. For now however, we turn to evidence of how biology appears to have solved control of such structures.

### 2.4.3 Bio-inspired and learning approaches

#### 2.4.3.1 Forward modelling

A predictive *forward model* of the system under control is widely employed as a component in control engineering (Atherton 2006; Levine 1996). Given a current state and a set of control signals it makes a prediction of the resultant end state. Although seemingly of less utility than an *inverse model* – which can supply the set of control signals required to move from a specified start and end state – it nevertheless finds a considerable range of uses. These include improving state sensing through Kalman filtering (Balakrishnan 1978; Wan & Van Der Merwe 2001; Welch & Bishop 2006), delay compensation (Mehta & Schaal 2002), Smith predictors (Smith 1959; Franklin et al. 2002), feedback error learning (Shibata & Schaal 2001) and optimal control where a forward model enables exploration to locate motor plans that minimise a cost function (Todorov 2004). It can of course, as recently discussed, also be used in classical closed loop control if it can be expressed as a mathematically tractable transfer function. However, a forward model may be implemented in other ways - such as a trained neural network or a physics-based simulation - and is often expressed thus in system designs as an unspecified ‘black-box’. For many controlled systems a forward model is

significantly easier to implement than an inverse model where derivation methods such as inverse kinematics (Atherton 2006; Levine 1996) cannot easily handle redundancy (numerous potential solutions). Indeed, one application of a forward model is in fact to provide an error signal for the training and correction, over time, of an inverse model (Demiris & Meltzoff 2008; Wolpert & Kawato 1998).

The widespread utility of the forward model in control engineering and robotics is one reason that the existence of a neural correlate in the motor centres of animals has often been championed (Miall & Wolpert 1996; Kawato 1999; Wolpert et al. 1998; Miall 1998; Wolpert et al. 1995; Flanagan et al. 1999; Webb 2004). The forward model is often proposed as a role of the human cerebellum (Blakemore et al. 2000; Wolpert et al. 1998; Miall 1998). Empirical evidence for this includes the clear physical presence of neural connections implementing motor efferent copy (Blakemore et al. 2000), a ubiquitous element of forward control systems in engineering. The presence of predictive models is also strongly indicated by the fact that effective reaching movements can be shown to be generated and performed faster than if any feedback mechanism were driving them (Desmurget & Grafton 2000). Other studies also suggest that the position of eye saccades tracking an unseen reaching movement reflect the output of a state predictor, rather than the actual position (Ariff et al. 2002). Stabilising grip force adjustments suggest a predictive ability via an internal model of motor apparatus during arm movements (Flanagan & Wing 1997). Furthermore, Kalman filter-like corrective mechanisms (which contain a forward model by definition) are implied in a number of phenomena including the flash-lag effect, (Nijhawan 1994; Eagleman & Sejnowski 2007), the cutaneous rabbit illusion (Kilgard & Merzenich 1995), the auditory continuity illusion (Grossberg 1995) and phonemic restoration illusion (Grossberg & Myers 2000). Existence of optimal control mechanisms (Todorov 2004), such as Kalman filtering, is evidenced by the velocity profiles of eye saccades and reaching hand movements (Collewyn et al. 1988), a bell-curve shape predicted by a cost function minimising endpoint variance under signal proportional motor neuron noise (Harris & Wolpert 1998; Tanaka et al. 2004).

We therefore suggest that if a form of forward model of this complex robot can be constructed, it may prove of significant value as a component in a overarching controller architecture for the robot that requires motor planning, sensory correction and delay compensation (see *Chapter 6*). However, as discussed, to derive the inverse

model for control via classical methodologies is very problematical due to the nonlinear complexity and high redundancy of the structure. Much of this research is therefore concerned with techniques (such as learning) to acquire what could be viewed, in system terms, as a black box inverse model (see *Chapter 4*). The availability of a predictive forward model will therefore also aid significantly in achieving this.

#### **2.4.3.2 Muscle-based control**

Newer control theories have emerged over recent years that focus particularly on the specific control opportunities presented by muscle-driven, compliant actuation rather than treating a musculoskeletal structure as a generic control structure of high complexity and dimensionality.

The Equilibrium Point (EP) Hypothesis (Feldman et al. 1998) was developed in response to the apparent paradox that observed posture stabilising reflexes should also prevent voluntary movement (Holst & Mittelstaedt 1950). EP hypothesis postulates that, reflexes could be considered as, not hard-wired responses, but rather, tuneable mechanisms. In this configuration, motor efferent copy could in theory be employed to drive stabilising reflexes to reset around a new posture defined in muscle length space, causing the change in posture to come about solely under the influence of these updated reflexes seeking their new equilibrium point (Feldman et al. 1998). Evidence cited includes the predicted force-length relationship observed in cat muscles (Matthews 1969). However, the hypothesis has been disputed as over-simplistic (Gottlieb 1998), citing evidence that motor control is acquired gradually through the development of internal dynamic models (Hinder & Milner 2003), and that measurements in further studies have not matched the predictions of EP (Lackner & Dizio 1994; Gomi & Kawato 1996; Gottlieb 1998). Furthermore, in a practical EP-based controller there is also the prerequisite of the new, target posture to be described fully in joint or muscle space (Gu & Ballard 2006), however, for a complex robot with high redundancy it is by no means clear how this is to be acquired.

An alternative muscle-based approach is suggested by a growing body of compelling empirical evidence from biology strongly suggesting the existence of muscle synergy-based modular control (Giszter et al. 1993; Kargo & Giszter 2000; d'Avella et al. 2003; Hart & Giszter 2004; Cheung et al. 2005; D'Avella & Bizzi 2005; Hart & Giszter 2010; Roh et al. 2011). A synergy here is defined as a fixed and distinct muscle activation

pattern distributed between its participant muscles and driven as a single unit by a control signal. These biological studies suggest that effective control of seemingly highly complex structures such as the bodies of frogs, cats or humans is, in fact, achieved largely through advantageous, co-evolved natural dynamics combined with a small set of relatively simple signals each activating a selection of precise muscle groupings (synergies) (Cheung et al. 2009; Ting & Macpherson 2005; Ma & Feldman 1995; Bizzi et al. 2008; Li et al. 2008). These significant findings suggest that if effective synergy patterns could be located for the biomimetic ECCERobot, then a limited set in simple weighted combinations might be similarly sufficient to produce effective movement under relatively elementary control.

#### 2.4.3.3 *Trial and error learning*

The effectiveness in humans of trial and error learning is readily apparent to the layman and is commonly formally implemented in learning algorithms as *reinforcement learning* (Barto 1995; Sutton & Barto 1998). Here, a binary or graded reward signal indicates success or failure of an action, or sequence of actions. The task of the learning is simply to adapt behaviour to obtain, over time, the largest net reward. This principle results in an increasingly focused search towards the best solution with little or no prior knowledge of its final form. However, whilst effective for simpler discrete problems, in the temporal control field such algorithms have proven in practice difficult and slow for high dimensional problem spaces and temporal sequences of actions. This “curse of dimensionality” (Bellman 1954) comes about because the computation and data requirements increase exponentially with the problem state size (Moore & Atkeson 1995; Peters et al. 2003). Since muscular control of a complex musculoskeletal body falls within this category it would thus not appear a suitable candidate for reinforcement learning based control.

However, in apparent contradiction, there exists significant evidence in neurobiology strongly suggesting that a good correlate of reinforcement learning *does* exist in motor learning through the selective release of the neurotransmitter dopamine (acting as a “reward”) to strengthen recently active synapses (Schultz 1998; Schultz 2002; Izhikevich 2007; Chorley & Seth 2011). This opens the possibility that biology may have evolved additional mechanisms to sufficiently simplify the control problem from a generic, high dimensional, non-linear structure to one that is amenable to control

acquired through a form of dopamine-based reinforcement learning that lies plausibly within the brain's abilities to acquire. Muscle synergies and evolution of amenable natural dynamics may be examples of such mechanisms.

#### ***2.4.3.4 Morphological computation and natural dynamics***

Just as conventional robots have been designed for classical control, so these theories postulate that biological bodies have co-evolved precise and subtle biomechanics to be as useful and amenable as possible to a brain-like CNS controller (Pfeifer & Iida 2005; Pfeifer et al. 2007). This approach mitigates the need for a highly advanced controller by co-evolving a more controllable body. The most cited example of this is the phenomena of passive walking (McGeer 1990; Hitomi et al. 2006) where the natural dynamics of an entirely unpowered set of legs allow it to walk unaided down a gradual slope in a natural and effective manner. Since the ECCERobot is based closely on human construction there may therefore be value in considering approaches that leverage natural dynamics. An extension to these ideas are control theories where control itself can "emerge" from these dynamics, driven by reinforcing information flows between the environment, reflexes, motor signals and proprioception (Der 1999; Te Boekhorst et al. 1999; Lungarella & Sporns 2006; Pfeifer et al. 2007; Gravato Marques et al. 2013).

#### ***2.4.3.5 Neural networks, evolutionary algorithms and spike-timing plasticity***

Animals control their complex, compliant bodies superbly not through formal algorithms or fast search but using richly connected neural networks in their brains and spines. It is therefore natural to look to directly ape this approach through simulated "brains". A partially bio-inspired approach is to train conventional artificial neural networks to learn a non-linear control function, often using evolutionary algorithms to search in connection weighting space for the "fittest" solution (Beer 1995; Cliff et al. 1993; Meyer et al. 1998; Bongard 2000). Success depends heavily on the size (dimensionality) and shape of the fitness landscape and designing an appropriate fitness function or functions for solving more complex problems can prove very difficult. An alternative is to construct more biologically-accurate spiking neuron simulations - including delays and plasticity - of indicated brain regions such as the cerebellum (Kawato & Gomi 1991; Izhikevich 2007). However, whilst relatively small simulated spiking networks have demonstrated success in solving easier problems or

controlling simpler robots (Luque et al. 2011; Carrillo et al. 2008), the complexity, connectivity knowledge, and network density of that would be indicated for the control of such a complex body as the ECCERobot are almost certainly beyond current neuroanatomical knowledge and simulation power. We therefore conclude that a heavily “brain-based” simulation controller is not a viable option, for the present at least.

#### **2.4.3.6 *Optimal Control***

The adaptation of control such that the system minimises the value of a specific *cost function* is known generally as *optimal control* (Todorov 2004; Wolpert et al. 2001; Harris & Wolpert 1998). Use of a particular cost function as a driver to modify behaviour can result in associated emergent characteristic behaviours, therefore observing these in nature can lead to inference of the underlying cost function, providing, in turn, a clue to designing better control. We will therefore consider in the next section of the review the evidence of optimal control in human motor control in order to potentially exploit the use of cost functions in developing control of the ECCERobot.

#### **2.4.4 Conclusion**

To fulfil the need for an effective muscle-based motor planner (or inverse model) the evidence suggests that conventional classical control or planning search approaches are unlikely to withstand highly complex and very high dimensional control subjects and that learning-based, bio-inspired approaches hold the most promise for controlling biomimetic structures. Action discovery approaches exploiting natural dynamics and compliance are favoured over precision trajectory planning ending at a fully pre-specified goal state. We therefore propose that muscle-based control techniques and reinforcement learning best merit further investigation along with exploitation of optimal control through identification of appropriate cost functions. In the final section of this review we therefore revisit and critically review this subset of control approaches in greater detail, with a view to selecting for investigation a combination with a strong mix of novelty, promise, and interest.

## 2.5 Bio-inspired evidence underpinning control approach selection

### 2.5.1 Introduction

We consider, in some detail, evidence for success of those bio-inspired approaches identified in the high-level review as demonstrating the greatest potential for control of the ECCERobot.

### 2.5.2 Standard Reinforcement Learning

As discussed above, reinforcement (RL) learning (Sutton & Barto 1998) is a general trial-and-error learning technique in which the task of the learning is simply to build, over a period of trials, a policy that is able to consistently select, for a series of presented problem states, the actions that will that lead to the most accumulated reward over time. Selecting when and how much reward is issued is therefore a critical element in constructing effective RL.

The reward issued, or punishment (negative reward) is attributed (attached) to the combination of action and states (pre and post-action) that led to its issue, where it is added to that already accumulated. The immediately preceding state-actions (the eligibility trace) may also be rewarded, in a decreasing scale, in order to build rewarding paths through state space. In order to avoid simply directing action selection towards those most used to date, the policy generally employs, not the accumulated reward, but the average reward issued per past selection of the state-action pair. This is known as the *value* of the state-action, usually denoted  $Q$ . This step is generally referred to as the policy evaluation. The change in policy reflecting an update in  $Q$  values is referred to as the policy update step.

A policy that always selects the highest value actions is referred to as “greedy”, however, short term gain may not lead to highest reward over time hence a standard refinement to RL is to balance greedy selections with exploration of alternative state-action space which may ultimately provide greater reward. For example, the  $Q$ -value may be used to set the *probability* of selecting that action, this allows seemingly less promising routes to be occasionally trialled. Another more fundamental issue is encountered in problems with a high number of micro-states, or continuous state spaces such as control of a real robot such as ECCERobot. As no two states measured are identical, this creates a problem in building reusable state-action pairs. This may be addressed using a state estimation function that attempts to increase the state

granularity or eliminate redundant or less critical dimensions, however this is rarely a simple problem.

Much research in RL has focused on extending these elementary approaches by adding sophisticated methods that act, also through learning, to refine the policy function itself. Examples include the *actor-critic* approach, which refines the policy parameters while the policy iterations are in progress by judging its success; and *temporal difference* (TD) learning which attempts to intelligently pre-populate its set of Q values using estimates from a function (itself adjustable through learning) that attempts to predict upcoming rewards. These estimates are then incrementally updated with evidence (real rewards) from actual trials conducted. It is largely these features that act to set reinforcement learning apart, through its ability to minimise what is termed *regret*; the favouring of rewarding actions in the short term leading to the loss of greater return later.

RL thus appears to be a proven generic learning method that can be effectively applied to motor control via these techniques. It is also an attractive theory to account for biological motor learning as it does not require repeated identical trials nor any explicit representation of desirable goal states, both of which are hard to come by in the real, noisy world.

However, in practice, although these approaches have proven successful in lower dimensional or discrete state-action spaces, for more complex control subjects with more than 5 or so degrees of freedom within a continuous state space, the resultant explosion of micro states necessitates the use of approximation functions that cause both significant performance issues – the computing cost rising exponentially - and convergence issues for the generic forms of these algorithm ( Peters et al. 2003).

Nevertheless, as discussed, although a biological body as a system to be controlled appears well beyond the point where standard RL becomes challenged, there is clear evidence that RL-like, reward-based approaches are indeed employed by the brain in motor learning. For example, patterns of synapse-strengthening dopamine release often appear to mimic, in both amplitude and timing, reward signals expected to be observed for RL methods (Schultz 1998; Schultz 2002). Although the exact and complete role of dopamine is disputed (Redgrave et al. 2007; Friston et al. 2012), a strong influence on motor control is not. Another source of empirical evidence for RL

at work in motor learning comes from the study of the characteristics of smooth, efficient human movement, which show evidence of optimal control (Todorov 2004) in the velocity profiles of movements such as eye saccades or reaching (e.g. Collewyn et al. 1988). Although the underlying cost function was believed to be minimisation of jerk (Suzuki et al. 1996; Breteler et al. 2002) this theory has since been superseded by a cost function minimising endpoint variance in the presence of amplitude-related motor neuron noise (Harris & Wolpert 1998). In other words, selected movements are those most reliable over repetition, a very clear benefit to the subject. Reinforcement learning appears to be a mechanism that can deliver this, namely, in unreliable or noisy systems RL will, over repetitions, inherently favour the most reliable solutions as they will accrue the most reward (Wolpert et al. 2001). The observation of these characteristic profiles therefore supports the presence of RL in motor learning since this inherent shift during learning towards optimally reliable, smooth (jerk-free) movement is not a feature of competing control theories such as equilibrium point hypothesis (Rosenstein et al. 2006).

Therefore, in choosing a control approach for the ECCERobot, we are presented with a contradiction. RL-like mechanisms appear well indicated in motor control biology, yet control of the body in a continuous space appears beyond the learning abilities of conventional RL algorithms. If we accept the former, then two possibilities to resolve the latter present themselves; firstly that conventional RL can be refined or developed further to handle much higher dimensionality, or that there is some other feature of the body or brain that is acting to simplify the control problem sufficiently, for example; increasing the linearity of the system response or reducing the dimensionality of the problem. We therefore investigate both of these possibilities, considering first the availability of high-dimensional RL techniques before moving on to the possibilities for control simplification through the approaches of morphological computation and muscle synergies.

### **2.5.3 Reinforcement learning for high dimensional state spaces and humanoid robotics**

Some success with learning to control high-dimensional systems in the real world, such as humanoid robots, has recently been demonstrated by sophisticated techniques that focus on optimizing the policy update step (Peters et al. 2003; Theodorou et al. 2010) which have demonstrated control of robotic systems, such as (conventional) humanoid

robots in swing up or balancing control tasks, striking a baseball with a hydraulic muscled arm and robot weightlifting (Kober & Peters 2010b; Peters et al. 2003; Peters & Schaal 2004; Schaal et al. 2004; Peters & Schaal 2008; Kober & Peters 2009; Kober et al. 2010; Kober & Peters 2010a; Theodorou et al. 2010; Schaal et al. 2003; Rosenstein et al. 2006).

However, although demonstrating a significant improvement in applying generic RL to high dimensional control, much of this work has focused on solving a difficult, but highly specific task, such as the swing up of a 2 joint actuated robot arm (Rosenstein et al. 2006). If the task parameters are varied even slightly after training then, unless the problem can be meta-analysed, learning must often be re-commenced (Kober et al. 2010). Relatively little work is focused on controlling or exploiting the features of explicitly biomimetic structures, such as compliance, although there is some focused on controlling muscle-based robots, which simply treats them as difficult control subjects (Peters & Schaal 2008). RL-based control of multi-muscle structures with a comparable level of redundancy as the ECCERobot is also relatively unexplored. Note also that although the terms *synergies* and *primitives* are commonly employed in describing these algorithms, they are generally used to refer to movement primitives which can be temporally chained together to form a larger motor plan (e.g. Rosenstein et al. 2006; Gu & Ballard 2006). This usage is significantly different from the biomechanical terms; muscle synergies and motor primitives, which refer specifically to weighted co-activations of muscles that evidence suggests provide control advantages in animals. In summary, although these sophisticated algorithms for generic high-dimensional systems are at the leading edge of RL research there is little evidence as yet that they will prove applicable to structures such as the ECCERobot in the near future.

#### **2.5.4 Morphological computation**

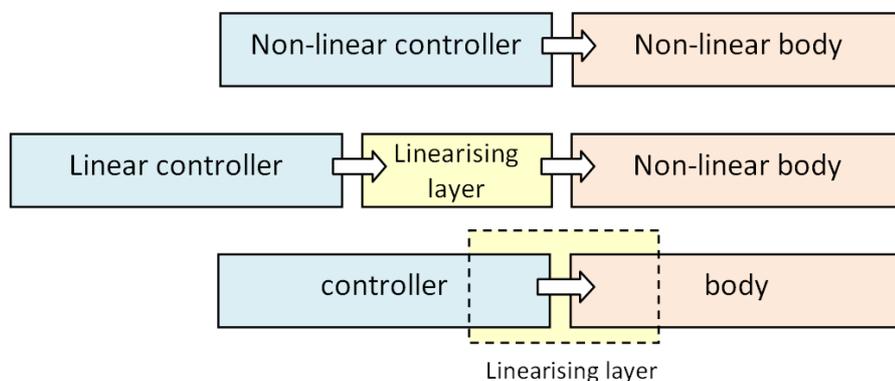
For musculoskeletal structures, locating the set of muscle signals to achieve a goal appears an extremely complex task due to the dimensionality, redundancy, compliance and the nonlinear, dynamical mapping from muscle activity to movement. However, in contrast to the class of RL problems discussed above, both morphology and mechanical structure has co-evolved in biology alongside the controller itself. This opens another route to easier, better control; make the body more amenable to simpler control, for example by the tuning of natural dynamics and compliance (Pfeifer & Bongard 2007). A well known example of this principle is illustrated by passive

dynamic walking (McGeer 1990), and it has furthermore been demonstrated by the natural walking motion of an unstable biped robot, which can be relatively easily stabilised by RL-driven parameter adjustment (Hitomi et al. 2006). By contrast, developing stable walking on a conventional humanoid robot, such as Asimo, has proven a very significant undertaking (Choi et al. 2004; Erbatur & Kurt 2009).

An important variation to this control approach also exists. In this scenario, just a elementary controller is available that is only able to control simpler structures with greater linearity and lower dimensionality. The relatively elementary control signals it produces are amenable to unsophisticated learning, such as simple RL. To allow this controller to succeed, an extra intermediate layer is introduced between the controller and the potentially complex, non linear body. The task of this layer is to manage or massage the “tricky” aspects of the body to offer an interface that can accept simpler and fewer control signals and respond more as if it were a linear control problem. Figure 4 (centre) schematically illustrates this approach.

It is important to note that the intermediate layer, whilst functionally distinct, may be physically implemented within either the controller or the body, or both. Furthermore its parameters may be partly or wholly plastic, allowing for optimisation through use and learning (see Figure 4 lowest).

Strong evidence from studies of muscle synergies combining with natural dynamics during movements now suggests that this form of general architecture, using forms of



**Figure 4: Using an intermediate layer to linearize control**

*Top:* a non-linear body requires a complex non-linear controller.

*Centre:* Conceptually, a simpler linear controller can be substituted by the introduction of an intermediate linearizing layer

*Lower:* In practice, the intermediate layer may be physically implemented within either the controller or the body, or both.

intermediate systems to add linearity and reduce dimensionality (e.g. Berniker et al. 2009; Neptune et al. 2009), may be close to that implemented by the brain and body, thus tackling some of major issues that impede the use of less sophisticated RL to control complex bodies. What remains less clear is the location of the implementation of this intermediate layer and the degree of plasticity offered, indeed these may vary significantly between species. Nevertheless, this appears a promising approach and we therefore now review biological evidence around muscle synergies and the results of trialling this approach in control studies.

### **2.5.5 Muscle synergies**

A muscle synergy can be considered any co-activation of muscles that produces a net torque on a joint or a net force vector (Flash & Hochner 2005). However, in this context we specifically refer to muscle synergies as those co-activations that reoccur most distinctly and frequently and generally serve a particular role in a motor action. If each synergy can be driven by a single neural output and contains a distinct activation distribution pattern between its participant muscles then a limited set might be sufficient in weighted combinations to produce a wide range of movement under relatively simple control (Flash & Hochner 2005).

Studies of muscle synergies have in recent years been primarily conducted in frogs and humans, although earlier work has also included cats. A major boost to these studies in recent years has been the refinement of component analysis techniques that can accurately extract underlying synergies from compound electromyographic activity from multiple muscles, including the detection of synergies activated with different amplitudes or phase timing (Tresch et al. 2006).

In frogs, analysis of activity from all muscles of the hind limb has shown that every one of a wide range of studied movements could be generated from a combination of fixed weighting muscle synergies. Many of these synergies were found to be common across behaviours whilst others appeared for specific behaviours alone (d' Avella & Bizzi 2005). Similar findings have been made in other frog studies (Hart & Giszter 2004; Flash & Hochner 2005), notably that extensive synergy reuse occurs between swim, jump and walk behaviours and that the differences in behaviour can be accounted for

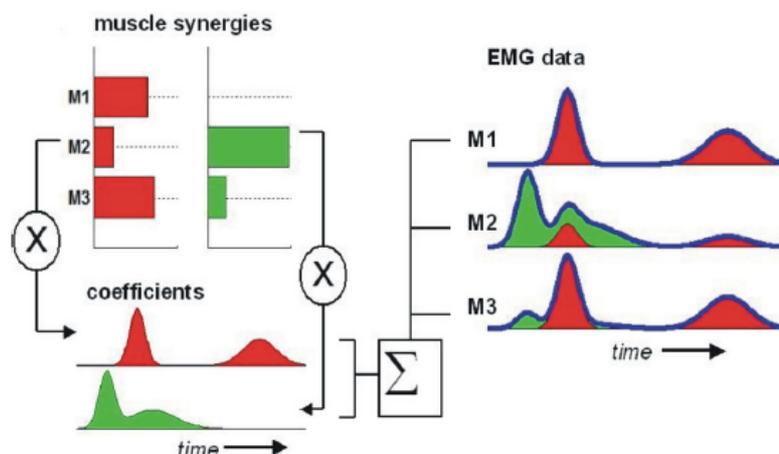
purely by variation in the amplitude and timing of the synergy activations (Bizzi et al. 2008).

Biomechanical modelling studies based on frogs have also succeeded in reproducing coordinated characteristic “wiping” movements across a range of starting positions using only fixed synergies adjustable only in gain and phase (Kargo et al. 2010). Interestingly this study found that, in contrast to earlier theories (e.g. Cheung et al. 2005), if the starting position is ignored then the behaviour resembled de-afferented frogs. The authors suggest this implies that proprioception is not used as part of a feedback modulated movement, but instead may simply act to obtain an initial estimate of limb position before an open loop motor plan is generated. Another modelling study where, in this case, synergies were selectively extracted from a detailed biomechanical model, were found to match those observed in real frogs (Berniker et al. 2009). Pertinently, these synergies had been specifically selected to exploit natural dynamic properties of the modelled limb, implying they generate the movements that the limb is naturally drawn towards.

Overall, all of these studies concluded that the frog motor controller has a modular synergy based organization, and that synergies exist which contribute to no single behaviour but are always found in cooperation with or modulating the outputs of behaviour specific synergies (D’Avella & Bizzi 2005). Synergies exploit natural dynamics of the limb (Berniker et al. 2009) although evidence also suggests that, for frogs, synergies are not learned but hardcoded as fixed modular “primitives” and that primary activation of each does not occur in the brain but is assigned directly as a single “module” to particular spinal interneurons (Hart & Giszter 2010; Bizzi et al. 2008). This appears to directly support the “intermediate layer” concept for control discussed earlier (Figure 4).

In human studies, evidence for shared, distinct modular muscle synergies is also very apparent, although where in the body or nervous system these groupings are primarily defined, and how they arise, is currently less clear cut.

Of these studies, perhaps most relevantly Cheung et al (2009) demonstrate that the seemingly complex muscle EMG signals captured during reaching can be accurately reconstructed from a combination of just a few fixed (time-invariant) muscle synergy patterns, if each is driven by a distinct, time-dependent, activating waveform. This



**Figure 5: Model of muscle pattern generation by a combination of muscle synergies (reproduced from Cheung et al, 2009)**  
 Illustration of reconstruction of recorded EMG signals from linear combination of time-invariant synergy patterns. Each is driven by a different time-dependent waveform acting as a coefficient. Both synergies (red and green bars) activate three model muscles; M1, M2, and M3. The waveforms generated by scaling the synergies with their time-dependent coefficient signals are summed for each muscle before comparing to the recorded EMGs (thick blue lines).

elegant finding is particularly relevant, therefore we reproduce it here (Figure 5), illustrating the principle at work. Note that, in this clear separation of simple driving signals from fixed weight groupings we see again a potential implementation of the “intermediate layer” architecture (Figure 4).

These reconstruction findings are supported by a number of other studies (d’Avella et al. 2006; d’Avella et al. 2008). For example, combinations of just five synergies, extracted during fast reaching movements, were found to explain around 75% of the signal data if appropriately scaled in amplitude and shifted in time. The same patterns were reproduced across different loads, postures or directions. Furthermore, it has recently been shown that the same set of synergies are simply modulated to correct movements when a target location is changed (d’Avella et al. 2011).

Similar evidence for explaining movements through synergies has also been shown in studies of human hand movements (Todorov & Ghahramani 2004; Weiss & Flanders 2004; Ingram et al. 2008). Of particular interest is direct evidence of the use of shared driving signals between synergies, shown by the commonality between signals driving the closing of each finger to form an overall grasping movement (Ingram et al. 2008). This also supports the existence of what can be termed “hierarchical” synergies (or “synergies-of-synergies”) where a “higher” synergy designed for grasping, recruits a

weighted pattern of “lower” synergies that control each finger in the correct proportion.

The potential application of such hierarchical synergies to explain full body movements is also demonstrated by studies of reaching movements involving both arm and trunk muscles (Ma & Feldman 1995). Here, one synergy was found to coordinate trunk and arm movements, leaving hand position unchanged, whilst the other produced inter-joint coordination to move the hand to the target.

Evidence from biomechanical modelling studies for synergy-based control in humans is also strong. Neptune et al (2009) used a complex musculoskeletal model of the leg to test if synergies alone were sufficient for effective locomotion. The synergies employed were extracted directly from measurements taken during studies of human walking. Only minor adjustments to the amplitude and timing shifts of the synergies were sufficient to generate the distinct characteristic phases of a step resulting in the formation of well-coordinated walking.

In a final example, we see the return of reinforcement learning to the table. In a study of simulated reaching, Fagg et al (2002) used a simplified and idealized musculoskeletal model of an arm and shoulder to demonstrate the acquiring of control by a simplified abstraction model of the cerebellum. Eight different muscle synergies were predefined amongst a total of six muscles. Simple reinforcement learning was then employed to learn, via trial and error, the sequence of synergy activations required to bring the arm to a specified target. This demonstrates how it is feasible for elementary RL alone, when coupled with a synergy-based layer, to succeed in learning the control of a bio-inspired musculoskeletal structure without resort to the advanced algorithmic complexities explored at the cutting edge of RL research.

Since empirical evidence of an intermediate layer in human control appears good, we will now also briefly consider specific evidence for its location and plasticity. In a study of cortical stroke victims (Cheung et al. 2009) the same synergies were extracted from muscles of both stroke-affected and unaffected arms. This suggests they must be constructed downstream of the damaged neocortex, the authors propose they may therefore be located in spinal inter-neuronal circuitries or the brainstem. A study of muscle signal interaction using Bayesian networks (Li et al. 2008) also suggests that synergies are defined outside of the motor cortex, but that they are not hardwired but

emerge causally (e.g. Hebbian learning) from correlated interaction between motor neurons or interneurons in networks in the spine. Thus, the neuron firing that drives one muscle can depend upon firings of neurons driving others; the authors refer to these as "dependent synergies". However, it is not clear whether these networks crystallize to form distinct patterns via action-reward mechanisms, such as spinal dopamine receptors (Lapointe et al. 2009), or by more unsupervised mechanisms in the manner of Hebbian learning .

Another indication of whether synergy groups are primarily learned or pre-wired is to consider how similar they are across subjects. Here, the evidence is conflicting. Ting and McKay (2007) claim that significant observed synergy variation across subjects in both number and composition clearly implies that they form through adaptation. Yet Cheung et al (2009) observe that the synergies they identify reoccurred with great regularity across subjects. The truth may lie somewhere in between, namely, that while the base hard wiring for the lowest synergies and reflexes is in place, they are harnessed via higher level synergies which form sufficient plasticity to cope with a range of natural variety of the bodies under control, each undergoing an individual growth and learning experience.

Although the evidence for synergy-based control in humans appears compelling, we will consider some of the contrary arguments that have been made. Firstly, similar human studies exist of muscle-directed movements that could not be consistently broken down by a synergy-based analysis (Kutch et al. 2008; Valero-Cuevas et al. 2009). Unlike the fast-reaching and walking studies discussed above that suggested significant synergy usage, these tasks required precise control through high attention and visual feedback. It has been shown (through PET scanning) that, in contrast to more instinctive actions, such conscious motor tasks appear to recruit additional "higher" brain regions (Stephan et al. 2002). This therefore instils a doubt as to whether these can be considered a like-for-like comparison. Nevertheless, we suggest that the results at least imply that distinct synergy modules may not comprise the sole interface to the muscles, in humans at least.

Another objection is that, while synergies appear to explain the data for the particular tasks that have been studied, they may simply arise as consequences of the optimum solution to the specific control problem. This solution may have been arrived at by

other optimizing mechanisms, such as minimizing noise or other optimization criteria (Tresch & Jarc 2009). However, we would note that that distinct synergy reuse across tasks is widespread (Bizzi et al. 2008), suggesting that synergies are not tailor made for each task, and that synergy-like nerve groupings have been physically located in spinal circuits of frogs (Giszter et al. 1993), rats (Ganor & Golani 1980) and also cats (Drew et al. 2008) - an animal particularly known for feats of balance and coordination.

### **2.5.6 Control approaches exploiting natural dynamics and compliance**

We will return now to consider the scope for controlling structures more easily by leveraging their natural dynamics. Two approaches are identified that can act to locate and exploit these; action discovery and formal analysis.

#### **2.5.6.1 Action Discovery**

Action discovery approaches rely on the notion that more amenable natural dynamics will be favoured during learning if relatively simple control signals are employed and the search or learning is not over-constrained. For example, we would judge success by the arrival of a hand at a target object rather than stipulating what the movement through state space should comprise, as would be the case with classical control.

Two examples of methods that can function as action discovery approaches for musculoskeletal structures are reinforcement learning and genetic algorithms. These can certainly, in theory at least, be configured such that success can be judged entirely by the outcome, not the means. In practice however, secondary techniques are often required to attempt to avoid excessive “creativity” in the solution. For example, a target can be struck by random violent flailings of the arm – a solution that is easy to discover but unsatisfactory for other reasons. An example of a secondary technique used with RL learning for robots, is to begin learning from an approximate solution based upon an imitation of an observed human movement (Schaal 1999; Schaal et al. 2003). Another solution is to introduce secondary success criteria, for example; using minimal energy in movements. However, in practice, this often requires a challenging balancing act between criteria such that one does not dominate the other.

#### **2.5.6.2 Formal Analysis**

Besides action discovery, an alternative, more formal approach to employing natural dynamics has also been demonstrated by Berniker et al (2009). Here, a technique known as balanced truncation is used for model-order reduction to obtain a low (5)

dimensional model of a dynamical high-dimensional, multi-muscle simulation of a frog hind leg. The low dimensional model attempts to capture each of the dynamics most effective in altering the task variable, such as the frog's foot location in space. For each dimension, the most effective muscle synergy controlling the state in that dimension was identified. A controller employing only these synergies applied to the low dimensional model was found to perform almost as well as a full non-linear controller developed for the complete dynamical model, but was far faster to execute. Finally, it was found that the synergies identified were in fact a good match for those extracted from real frog movement. However, we note that this approach does not use any form of fitness or cost function based on the larger goal, for example, generating the most efficient swimming action for the frog. This task is left for the low dimensional controller activating the extracted set of synergies.

## **2.6 Selection of Control Approach for ECCERobot**

In conclusion, evidence suggests that conventional reinforcement learning alone will prove insufficient to overcome the dimensionality of a structure as complex as the ECCERobot physics model. The use of more complex advanced RL techniques appears a viable option, but offers relatively little novelty as it will treat the structure to be a generic n-dimensional problem, although the potential for leveraging natural dynamics remains of possible interest. On balance, it is debateable whether attempting to apply these techniques will add much value to the evidence base beyond demonstrating that they do, or do not, succeed with such a structure.

By contrast, the muscle synergy approach is of significant interest to research that is specifically concerned with the control of biomimetic structures. This is particularly true in the robotic field where musculoskeletal work to date has largely focused on the engineering challenge over the control one. Yet there is good evidence suggesting that, once effective synergies have been identified, the control problem is very significantly simplified and can exploit the morphological computation and natural dynamics aspects of the structure. Furthermore, in the modelling field, whilst there is sufficient work with musculoskeletal models to suggest that this approach may succeed, much of the work to date is frog-based, employing mainly isolated limbs with little work undertaken with full body models. Human-based models have tended to be either generic, idealized, muscle-based structures or detailed models of a very specific part of

human anatomy where synergies can be copied from analysis of real muscle data. There is no synergy-based work as yet targeted at control of biomimetic robots where effective synergies may turn out to at least resemble, if not reproduce, those of the real animal (due to the unavoidable gap in construction methods and materials). For example, the ECCERobot, albeit well-muscled with attachment points based on Gray's Anatomy, nevertheless has by necessity only a fraction of a human's musculature.

The question arises however, if this approach were adopted, how would effective muscle co-activation patterns be located? We note that a low dimensional representation of an action such as that proposed by Cheung et al (2009) is sufficient to describe real muscle-activation data during human reaching (Figure 5). Learning to use this form of reduced parameter space to uncover the most rewarding muscle weightings and driving signals appears potentially within the reach of conventional reinforcement learning. Although not the only learning technique capable of this, as we have discussed, RL brings other favourable aspects with it. Apart from being consistent with some aspects of brain function, it can also employ action discovery to exploit useful natural dynamics (morphological computation) and can also, in theory, lead to optimally reliable reaching-to-target movements when under conditions of signal dependent noise combined with Monte Carlo trialling (Sutton & Barto 1998); i.e. randomised trial repetition generating a probability distribution of outcomes. We may then look for emerging elements of human-like smooth movement through the emergence of signature bell-curve velocity profiles.

Furthermore, since we are effectively compelled to employ either a relatively slow-running model (due to its modelling complexity) or, ultimately, a slow and vulnerable real robot, the ability of RL to learn cumulatively from every trial is also a valuable feature.

Nevertheless, this approach still presents a number of potential issues. Firstly, although a relatively lengthy behaviour (e.g. a reaching action) in the timescale of seconds might be describable by relatively few parameters by employing a sustained activation of a simple combination of fixed muscle synergies, it must still be attached to a state to obtain state-action pairs to which reward can be assigned. Since the complete kinodynamic state space (incorporating position and velocity) of this model is very large there is still a requirement for state estimation techniques (Sutton & Barto 1998).

A second issue is that in such large state spaces RL is likely to function significantly better as an improver of an approximate, weak solution than as a reliable bootstrap mechanism, shifting any behaviour towards the optimum region of solution space. If this proves to be the case then consideration must be given to how learning can be kickstarted into useable regions of solution space.

## 2.7 Control Target – robot or model?

Finally, we consider whether our selected approach should be developed against the physical robot or a modelled approximation, at least for preliminary investigations.

Although the robot remains the ultimate target for control, to commence investigations with a full sized, high powered and powerful robot brings significant practical issues to the fore, primarily that experimental control signals may cause excess wear and tear and even damage. One alternative is to employ a minimal robot test chassis, perhaps a single anchored arm and shoulder, for early investigations. However, this approach would have a significant impact on strategies that look to exploit full natural dynamics of the body whilst also needing to overcome control issues such as compliance-based oscillation of the body, a flexible spine that must be supported by muscle tension and highly unconventional structures (in robotic terms at least) such as fully floating shoulder blades.

We therefore argue that during the initial phase of exploring potential avenues for effective control, a detailed dynamic model closely approximating the *complete* robot, would prove of greater benefit. This would provide a convenient and realistic platform for trialling control approaches or for extended periods of offline learning or planning search. Such a forward model may also serve a second useful purpose an important component in an overall controller architecture requiring a forward model, offering features such as delay compensation and Kalman filtered proprioception.

Nevertheless, the goal of achieving control of the real robot remains important and potential transference of a proposed approach from the model to the physical robot must be given consideration whenever possible.

If a model is to be used, then consideration must be given to using existing available models, or biomechanical model-building tools. We find that although numerous biomechanical models of individual human body parts or small regions exist, both as

simplified/idealised forms and as detailed biologically-based musculoskeletal simulations, very few full-body human models exist and none of comparative musculoskeletal robots. Those that exist (e.g. *AnyBody*<sup>TM</sup>) are not designed as control platforms, but rather as medical or sporting tools and take as input real captured motions rather than the direct muscle activation signals we need. For the ECCERobot we require, if possible, a relatively fast simulation model, not of a human, but of a complex hand-built robot which, by necessity, is constructed with real materials and constraints as an engineering approximation to a human.

However, as discussed earlier, to construct a classical control mathematical model as a set of differential equations has already been shown to be near-impossible for this highly non-linear, high redundancy robot structure (Potkonjak et al. 2010). The structure may prove better suited, we would argue, to modelling in the kind of step-by-step approximation afforded by a constraint-solving physics engine. Here, on every time step, the solver repeatedly iterates the new state estimate towards one which better satisfies that set of constraints (e.g. joints) and forces (e.g. motor torques) describing the system at that point. The setting for the number of solver iterations is generally selected as a trade-off between performance and accuracy dependent on the application.

We therefore propose the use of a fast, modern physics simulation engine for the construction of a detailed physics-based model of a *complete* anthropomorphic robot, incorporating the potential to exploit full body natural dynamics and even plan and test interaction with sensed environment objects, which themselves may be modelled dynamically within the same physics “world” as the robot model.

## 2.8 Conclusions

To fulfil the need for a muscle-based motor planner (or inverse model) we argue that conventional engineering control or planning search approaches are unlikely to withstand highly complex and very high dimensional control subjects and that learning-based, bio-inspired approaches hold the most promise for controlling biomimetic structures. We therefore propose the design of a learning controller for discovering effective reaching actions through weighted synergies, drawing upon strong, recent evidence from muscle synergy research in frogs and humans; an

approach very little explored to date in robot literature. Since effective synergy patterns for a robot will be unknown, we propose to commence with simple reinforcement learning approaches intending that these muscle-coactivations will be encouraged to emerge, in particular those that aid linearization of the control. We also propose to draw upon optimal control theories to encourage the emergence of smoother, more natural movement by incorporating signal dependent noise and trial repetition.

Finally, in considering whether our selected approach should be developed against the physical robot or a modelled approximation we argue that, while exploring potential avenues for effective control, a detailed dynamic model closely approximating the *complete* robot, would prove of great benefit. This would provide a fast, convenient and realistic platform for trialling control approaches seeking to exploit natural dynamics of the full biomimetic structure or for extended periods of offline learning or planning search. The same model may also serve a second role as an important component in a predictive model based controller architecture, offering features such as delay compensation and Kalman filtered proprioception.

We conclude from a review of available full body models and musculoskeletal model building tools that none are fit for the purpose of an anthropomimetic robot controller. We therefore propose employing a fast, modern physics simulation engine to construct a complete physics-based model which incorporates actuation modelling, demonstrates full body natural dynamics and can potentially predict dynamic interaction (e.g. collision) with sensed environment objects.

## Chapter 3

# Developing a physics-based model of a complete anthropomorphic torso under compliant muscle actuation

---

### 3.1 Overview

In this chapter we present work undertaken to create a detailed, full-body, physics-based simulation model of one generation of the anthropomorphic ECCERobot (Holland et al. 2010; Holland & Knight 2006), known as the ECCERobot Design Study (EDS). The model was created for three main purposes.

Firstly, to investigate whether such an extensive highly dynamic structure can be usefully modelled within a standard physics engine, to produce a stable and comprehensive simulation running at a real time or better speed while incorporating the main features of interest.

Secondly, to provide a realistic, comprehensive test platform for developing effective control methods applicable to a whole body biomimetic robot with compliant actuation. In particular, we are interested in macro control issues or features that do not clearly manifest themselves in studies considering minimal models of only one or two joints (such as Wittmeier, Jäntschi, et al. 2011; Potkonjak et al. 2010).

Finally, to develop a model that could be integrated as a module into a control architecture, acting as a motor planning resource or a state-predictor within a delay compensation mechanism (Diamond et al. 2011).

This chapter will describe in turn the four main stages of producing the physics-based model. Note that where further detail is available from technical reports produced for the ECCERobot project these are included as appendices and referenced. Where detail is available in papers published by other institutions forming the ECCERobot team these are summarised and fully referenced.

The four main stages described are as follows:-

- 1) The capture of the robot morphology from the physical robot to create a static 3D CAD model in a standard format.
- 2) The analysis of the structure and migration process to create a passive (no-muscles) physics-based model that can be loaded into a standard physics engine.
- 3) The process to add a simulation of muscle-based actuation to the passive model including custom components requiring development for implementing elastic muscles, joint friction and wrapping muscle cables.
- 4) The testing and validation of the model, including the creation of a preconfigured “starting state” free-standing version of the model where the muscle lengths have been pre-tensioned to hold the torso indefinitely upright awaiting motor commands.

### **3.2 Capturing robot morphology to create a static 3D model**

The ECCERobot itself was constructed by hand using human anatomy as a guide and no upfront CAD modelling was employed – for example the polymorph bones are all hand moulded. The complete robot structure was therefore to be first reverse-engineered into a 3D modelling tool using an array of detailed photographs and measurements. Some examples are shown in Figure 6. A number of videos were also recorded demonstrating the action of all the robot joints and actuation for use in the third stage of modelling where the simulation of muscle-based actuation was added to the passive model.

Data capture was limited to the relative positions in space and approximate shapes and of the major components, namely the bones, motors, joints, pulleys, and muscle cable runs and attachment points.

The modelling tool Blender was employed to create the static model. This tool is not only very powerful but is open-source (a pre-requisite of the project) and includes the ability to export in the industry standard 3D COLLADA format which is compatible as an import format for most major physics engine implementations.

Each component was first approximately modelled by hand in Blender with reference to the close up photographs. The wider view pictures were then imported into Blender

as background images and used iteratively to adjust the relative position and orientation of each overlaid component until all relevant photos were consistent with the modelled version. Note that it is possible to precisely adjust the “viewpoint” coordinates in Blender until the original camera location is reproduced. These positions may then be saved along with the model. Note also that all photographs included a ruler to provide a good guide to correct scaling.

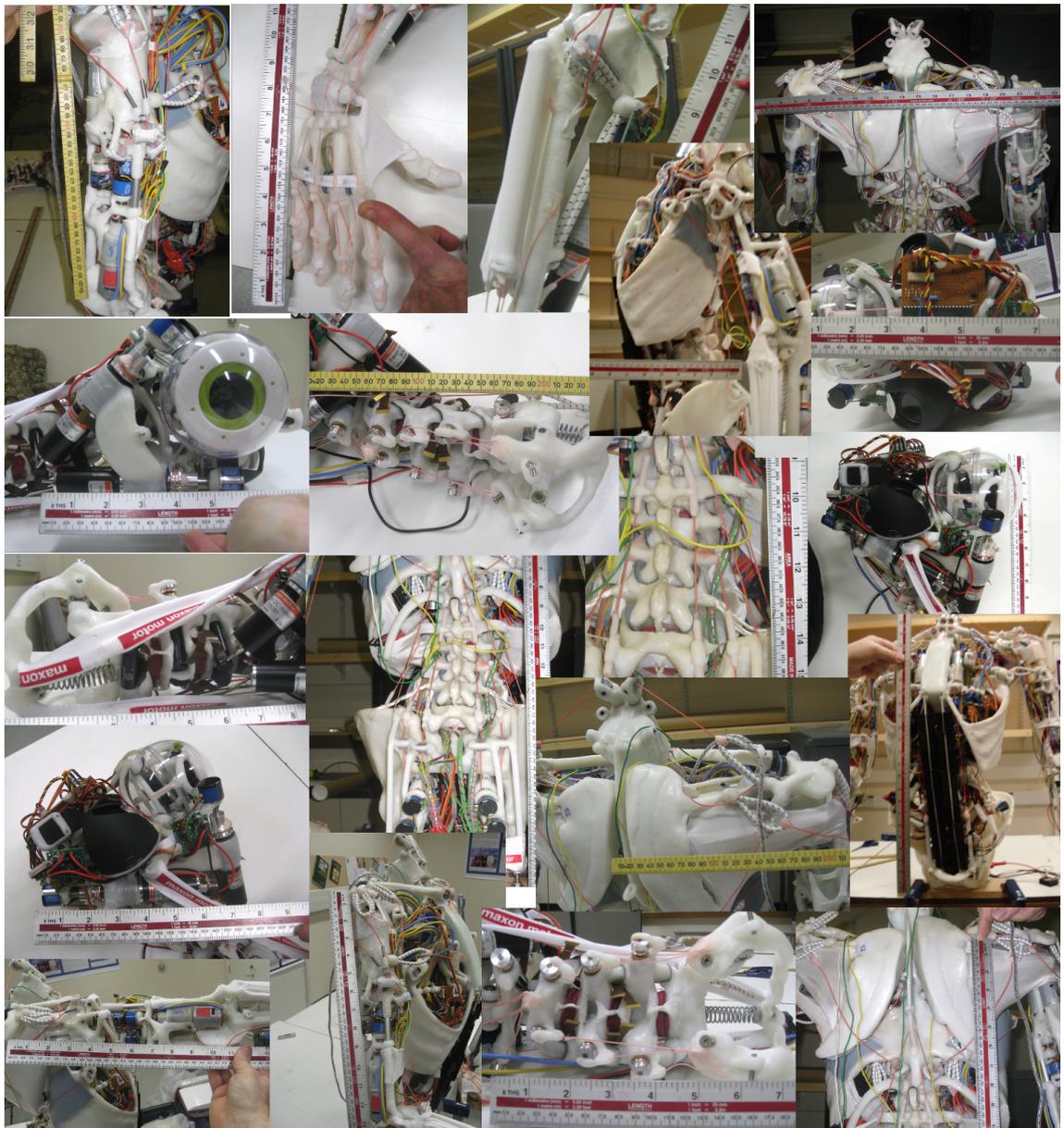


Figure 6. Selection of source material generated for reverse-engineering of ECCERobot model

Stages in the construction of the static Blender model are shown in Figure 7. The completed static model is fully detailed in Figure 8 and Figure 9.

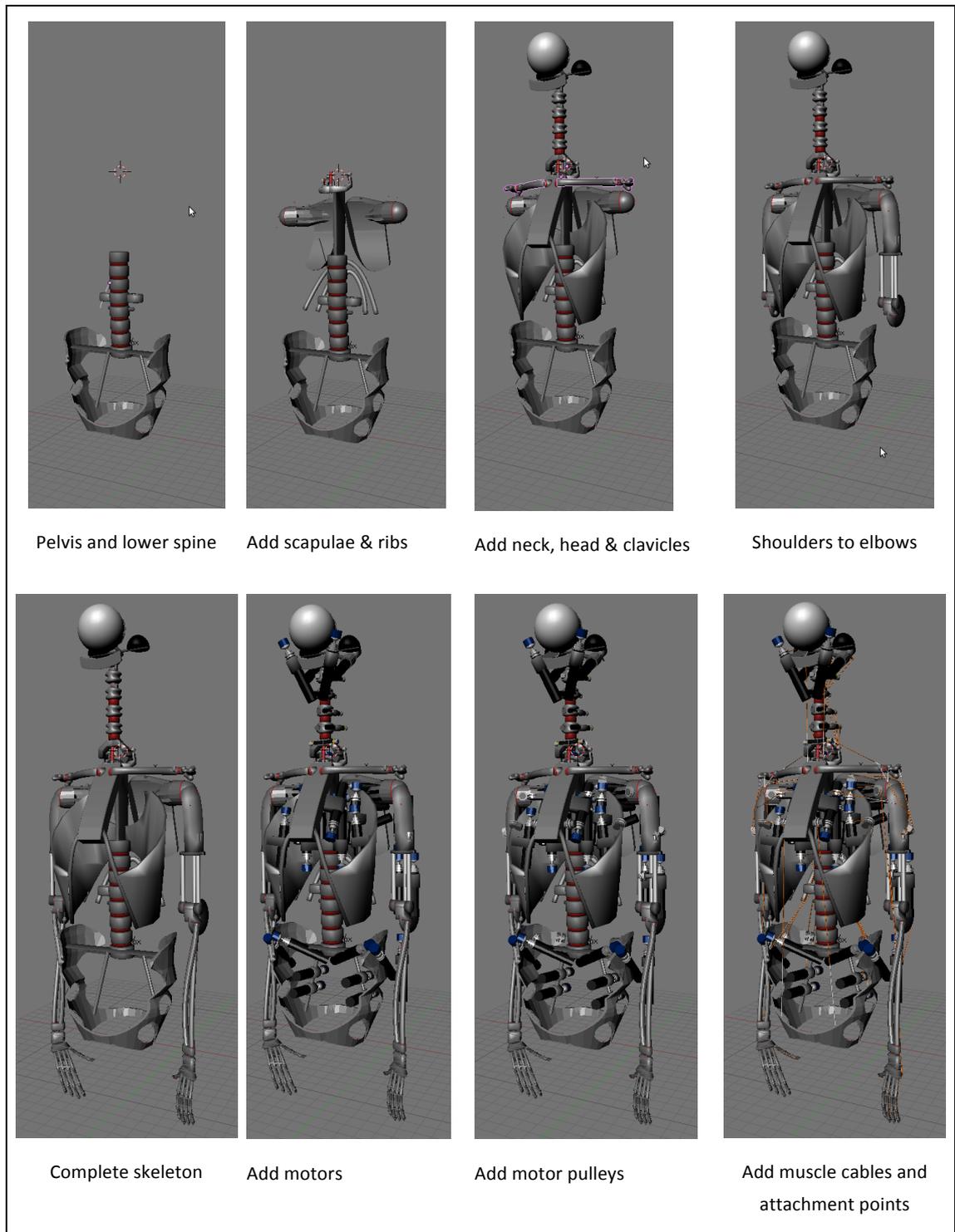


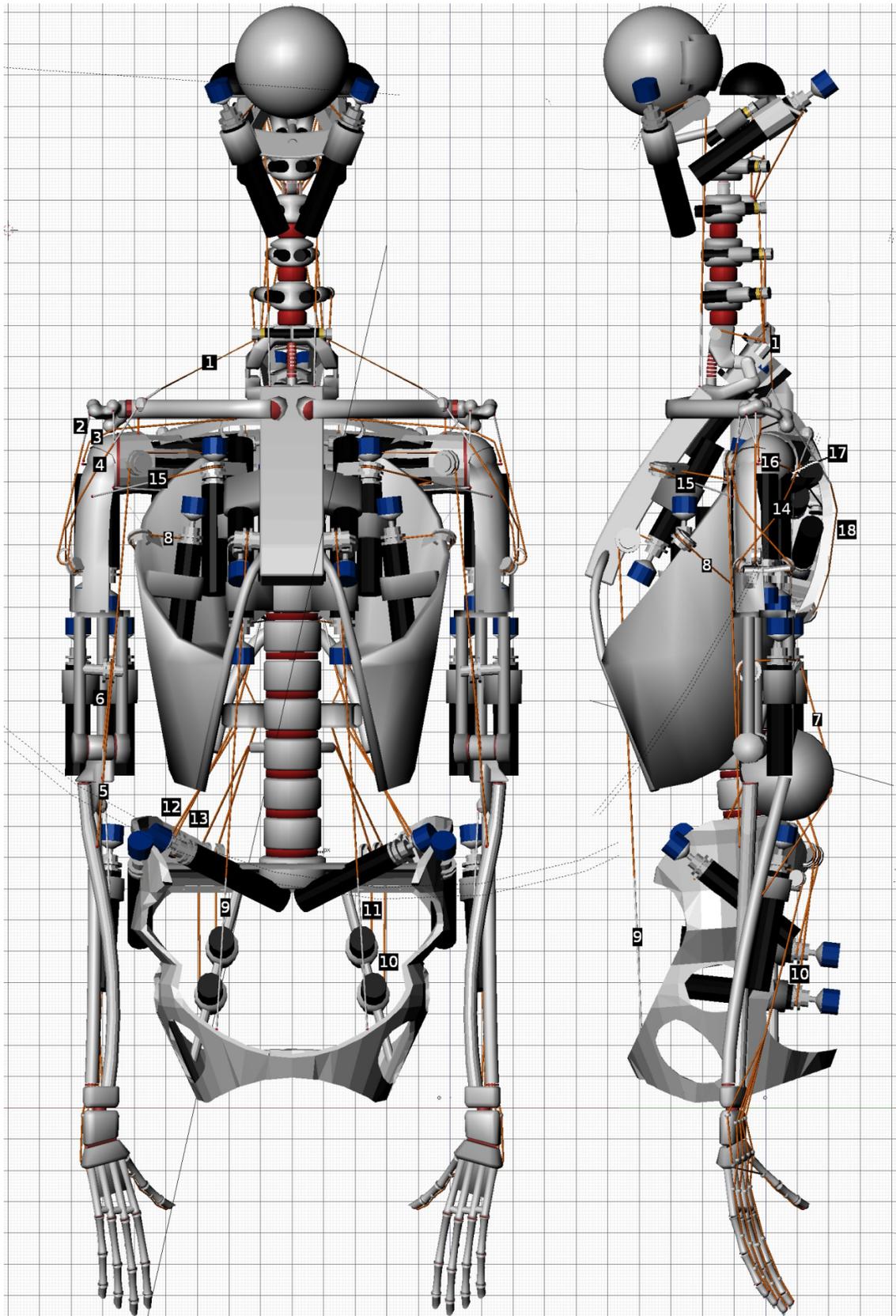
Figure 7. Stages in development of static model using the Blender tool

Further details of the capture process are provided in the technical report for the project deliverable pertaining to the modelling. The full report is attached following the thesis as Appendix II. It should also be noted that alternate capture approaches such as laser scanning and attachment point calibration using evolutionary algorithms were also trialled against a minimal test rig arm with some qualified success, detailed in (Wittmeier, Gaschler, et al. 2011).

### **3.3 Selection of physics engine for dynamic simulation of robot**

Employing an off-the-shelf physics engine offered practical advantages over developing a custom analytical model. These include the fact that support exists already for a range of rigid bodies and joining constraints and dynamic models can be constructed using existing standard modelling tools, 3D viewer libraries and file archive formats. Large complex models are a realistic option as performance is generally high since the engines are designed for real time game simulation and speeds are increasing through widespread adoption of GPU acceleration. Furthermore, they offer the substantial potential advantage for motor planning by offering direct integration of an environment with the modelled robot for such applications as collision-free motion planning. Indeed, later prototypes of the ECCERobot were designed to capture and integrate the live environment via a head-mounted MS Kinect sensor (Devereux et al. 2011).

To select the best tool a comparative review of features and performance was conducted by the project team of the leading physics engines (PhysX, Havoc, ODE, Bullet). The full review is available to view as Appendix II. From the review, the modern Bullet Physics ([www.bulletphysics.org](http://www.bulletphysics.org)) was selected as the platform offering the best combination of a flexible, extendible C++ based architecture, open source status and a fast impulse-based design. Impulse-based simulations are simpler – therefore faster – than constraint-based as net forces can be calculated as the sum of impulses over a short timestep of contributing bodies (Mirtich & Canny 1995). Constraints such as joints are implemented by issuing resisting impulses following constraint violation rather than solving for absolute constraint rules (Mirtich & Canny 1995). This is therefore fast, although potentially problematic as this post-hoc resistance means that constraints or joints can behave in an elastic-like manner or even give way under a large and sudden force.



**Figure 7A. Reverse-engineered static model of ECCERobot torso (Front and Side)**  
 See figure 7B on next page for back and top views, plus key to muscle names.

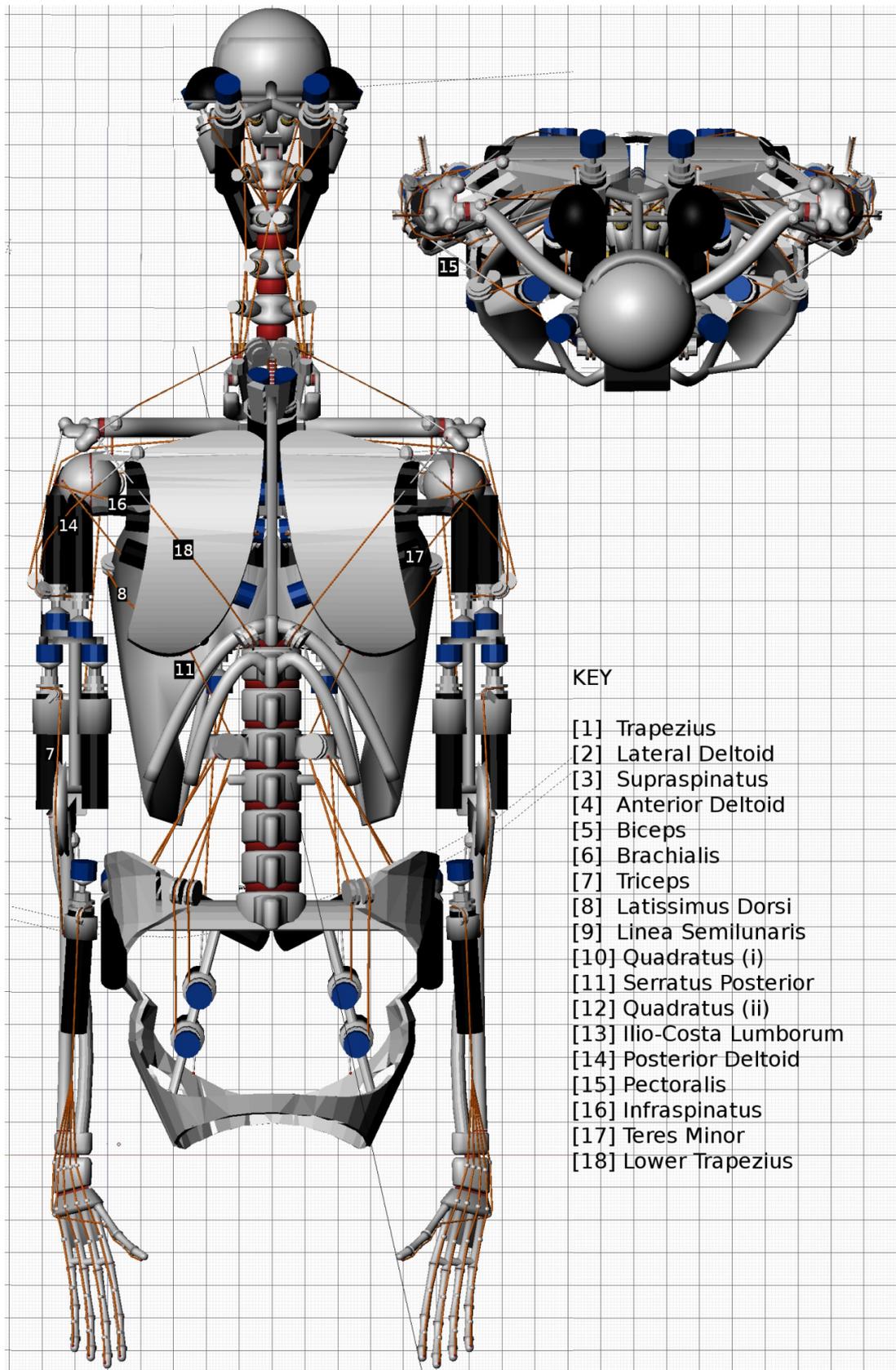


Figure 8B. Reverse-engineered static model of ECCERobot torso (Back and Top)  
See figure 7A for front and side views.

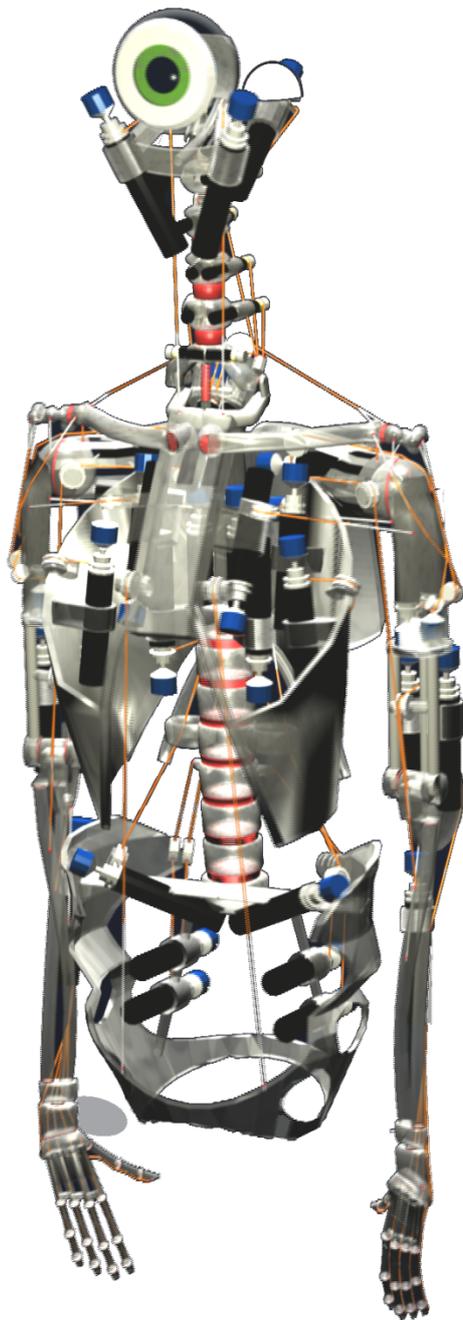


Figure 9. Textured rendering of static model of ECCERobot using translucent bones to show interior

### 3.4 Preparing for simulation through analysis of joints, constraints and other issues

Prior to commencing the physics based model a detailed preparatory study was made to collate and analyse every constraint requirement, then propose an implementation approach for each. A particular issue was whether the constraint could be satisfactorily addressed by the capabilities of the physics engine or whether a custom extension was required. Further details are available in the project technical report (Appendix II). A section of the analysis spreadsheet is shown in Figure 10.

	A	B	C
1	Picture	Name	BlenderName
26		Neck Base Hinge Constraint	NeckBaseHingeConstr
27		Neck Base Spring	NeckBaseSpring
28			
29			
30			
31			
32			
33			
34		Clavicle Sternum Constraint	ClavicleSternumConstr
35			
36			
37			
38			
39			
40			
41			
42		Clavicle To Clavicle Extension	ClavToClavExtConstr
43			
44			
45			
46			
47			
48			
49			
50			
51			
52			
53		Shoulder Joint Constraint	ShoulderConstraint
54			
55			
56			
57			
58			
59			
60			
61			
62			
63			
64			
65			
66			
67		Lower Spine Sponge Disc 1 to 8	LowerSpineSpongeDisc[n]
68			
69			
70			
71			
72			
73			
74			
75		Elbow Hinge Constraint	ElbowHingeConstrInner / Outer
76		Rotate Ulna Constraint	RotateUlnaConstr
77			
78			
79			
80			
81			
82			
83			
84			
85			

Figure 10. Screenshot of spreadsheet compiled detailing every constraint and proposed implementation

## 3.5 Creating a physics-based model of the passive structure

### 3.5.1 Instability in the physics engine

The complete structure of the robot is very complex for the physics engine to model whilst maintaining correct and realistic behaviour. The primary issue is not the number of bodies (which is low on the scale for such an engine) but the number of inter-dependent constraints that join them together in a single dynamic structure.

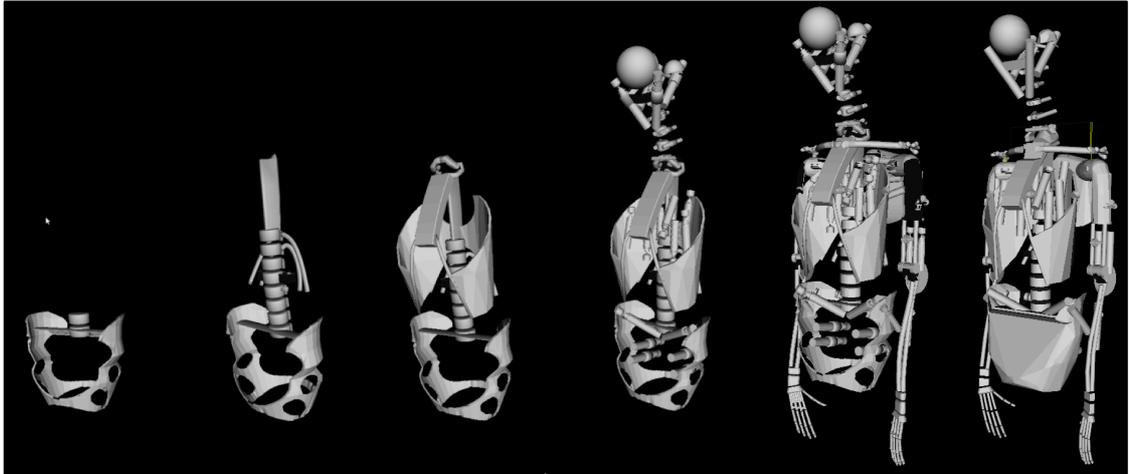
It is very easy for the engine to fall into a state which we will refer to as “unstable” where the bodies exhibit sudden and highly unrealistic, extreme behaviour as the impulses controlling its constraint mechanism appear to fall into regions of positive feedback. Typically this results in the model “exploding” or whirling its connected parts at high velocities.

To combat this we found that the best way is to proceed cautiously step-wise whilst constructing the model, ensuring that every body and constraint added is well behaved before moving on.

### 3.5.2 Building the model incrementally

The static model created of the robot (see Figure 8) was migrated to Bullet one body at a time and incrementally joined up with defined constraints to ultimately form a single passive (un-actuated) structure within the physics-based simulation. The result was a COLLADA file defining each body, its relative position and orientation and each joining constraint.

The migration was undertaken one step at a time by exporting from Blender via the standard XML-based COLLADA 3D file format and then adding the XML stanza generated to the gradually growing full model file, then testing this can be loaded. The constraint definitions are then added that attach the new body to those already in place and it is tested again. In order to allow the structure to stand while in this interim state a static (unmoveable) vertical pole was added behind the spine and the highest section of the structure so far was hung from it using a 6DOF constraint acting as a surrogate cable. The step wise construction of the model is illustrated in Figure 11. Examples of the COLLADA XML defining the model is shown in Figure 12.



**Figure 11. Stages in the migration of the static Blender model to a definition of the physics model in the Bullet engine**

The last two frames both show the completed model, first as full-mesh detailed bodies imported from Blender which will be used for the visual display of the model and secondly as a set of simplified collision shapes composed largely of primitives for maximum performance in collision testing, a common bottleneck area.



**Figure 12. Defining a physics based model as an XML file**

Examples of COLLADA-based definitions of a joint constraint and a rigid body composed of multiple collision shapes

### 3.5.3 Implementation of specific modelling issues

#### 3.5.3.1 Use of primitive shapes to speed collision detection

To significantly accelerate collision detection, custom mesh shapes exported from Blender were replaced in the model file wherever practical with one or more primitive bodies (cylinders, cuboids, spheres etc.) (see Figure 15) since an ad-hoc mesh shape requires testing by the collision detector of the relative location of every individual

face. Primitive shapes, by contrast, can be tested holistically. A single body can be constructed of multiple primitives, potentially overlapping in space.

### 3.5.3.2 *Defining body weights*

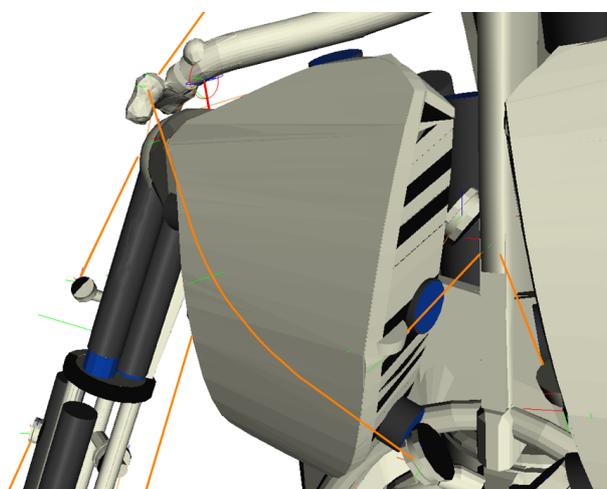
The physics engine, unlike Blender, requires a weight to be defined for each body. Bone weights were extrapolated from the material density and volume. Compound bodies, such as metal motors mounted on polymorph bones, were modelled by a single compound collision shape with the altered centre of mass and inertia tensor pre-calculated based on the combined mass and shape of each (Wittmeier, Jäntschi, et al. 2011).

### 3.5.3.3 *Modelling the spine vertebrae*

The upper (neck) and lower (back) spine sections are modelled by individual vertebrae (rigid bodies) which are joined with additional constraints acting as surrogate tendons to avoid S-curve collapses.

### 3.5.3.4 *Modelling bodies joined by inelastic cable*

A number of parts of the robot are joined solely by sections of inelastic kiteline, notably the construction where the “floating” shoulder blades (to which the arms are joined at the shoulder joint) are simply hung from the collarbones. These cable joints were modelled using 6DOF constraints (Figure 13). Note that these shoulder blades are ultimately held in place only by the muscle cables wrapping them (the next section will discuss how this was achieved in the model).



**Figure 13. Physics-based model: the floating shoulder blade and arm**

The shoulder blades hang from the collarbone (red constraint), held by wrapping muscle cables (orange)

The collarbones themselves are joined loosely to the breastbone alone by a 3DOF constraint, the other end hanging from the trapezius muscle cable. It is important to note that these structures are very hard to model analytically and lend themselves far more naturally to this class of physics engine, where it can be seen, once the muscle cable simulations are introduced to the full model, that this complex and free moving structure settles into a stable state with a subjectively natural pose. This may be a consequence of the close attention to detail in modelling the robot from human anatomy.

### ***3.5.3.5 Joint friction***

The implementations of static and sliding friction are fully detailed in the project publication (Wittmeier, Jäntschi, et al. 2011). Essentially, static joint friction is simulated using the joint motor feature of the Bullet physics engine to temporarily hold the angular velocity at zero. Sliding joint friction is simulated by applying an opposite angular impulse per timestep proportional to the angular velocity.

### ***3.5.3.6 Statistics of complete passive model***

Overall, the full robot model includes some 64 separate rigid bodies (independent moving parts or assemblies) defined by 246 collision shapes. There are 63 separate passive constraints (joints) and a total of 88 degrees of freedom in movement. This total includes details such as fully jointed fingers – although the later versions used for developing control omit these non-critical elements (unused in non-grasping motion planning) to obtain the significantly higher performance available by limiting the number of joint constraints.

## **3.6 Simulating the active structure**

To complete the model requires the addition of actuating muscles to the passive structure. We present an overview of the process here. As before, further details are available in the project technical report provided as Appendix II.

### **3.6.1 Approach for modelling of muscle cable forces**

The effects of the muscle cables acting on the body were “virtually” modelled by introducing additional impulses to the simulation during the callback function invoked by the engine at the end of each physics timestep. The forces were calculated by tracking the amount of kiteline currently unwound from the motor and comparing this

to the current distance to the attachment point. Any discrepancy was assumed to be taken up by the elastic shockcord and the tension force was then calculated via Hooke's Law (see again Wittmeier, Jäntschi, et al. 2011 for details).

### **3.6.2 Strategy for adding muscles incrementally to the passive structure**

As with the passive structure, muscles were added and tested one at a time to ensure stability is retained in the physics engine. To add a muscle the attachment points are exported from the static Blender model and used in a defining XML stanza that is added to the model file. This uses a custom format as such muscles are not supported by standard COLLADA. The import of the COLLADA file into Bullet was extended to interrogate these stanzas and add the hooks for each corresponding callback that will introduce the appropriate actuation impulses into the simulation depending on the model state.

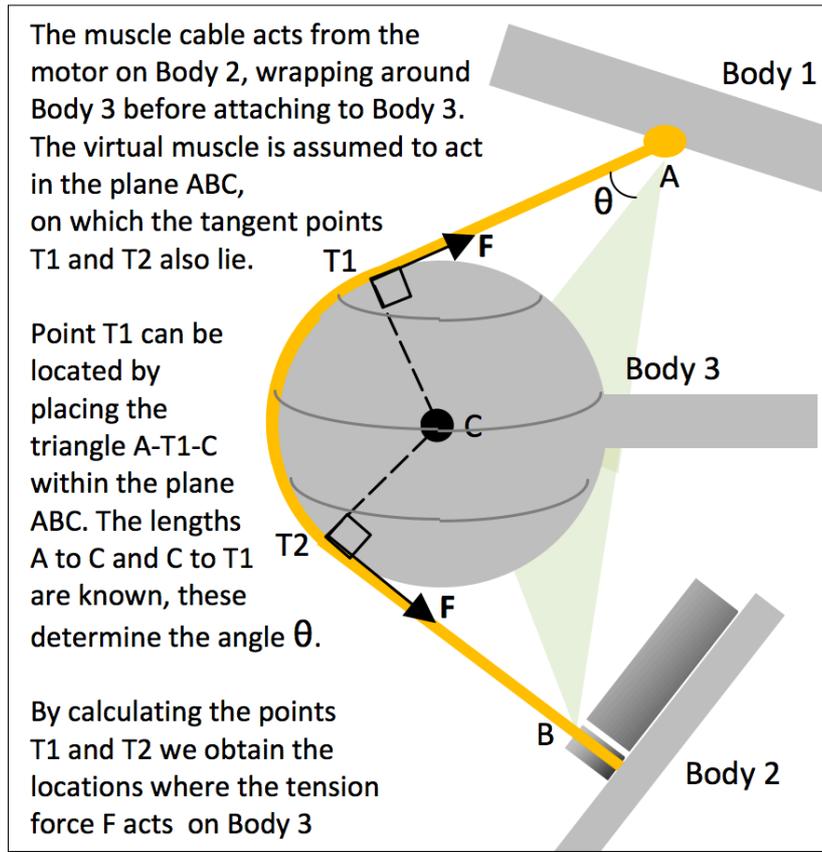
### **3.6.3 Modelling bodies joined by elastic cable**

A number of parts of the robot were identified in the review as joined solely by sections of passive elastic shock cord. These were modelled as unmotorised virtual muscles cables, as a 6DOF constraint (as used for inelastic cables) offer no elasticity.

### **3.6.4 Muscle cables that wrap bodies**

For this full-body model, a critical extension was added to the engine to simulate the effect of muscle cables wrapping around bodies. This is a vital element of the mechanics of the robot (aping the human body), certainly more so than the effect of the true pulleys in the structure, which primarily act to pinpoint the point at which a muscle cable should act on a body part. For example, free floating bones such as the scapulae are held in place by wrapping muscles and the shoulder joint in particular is actuated by several muscles that come from the scapula and pass around the shoulder ball joint before attaching. Without this feature the shoulder could not be actuated by the motors and the scapula would dislocate.

Initial attempts to simulate cables as colliding "soft" bodies were rejected as unstable. Instead a system of spherical virtual "pulleys" was introduced. These can be placed on any rigid body and the muscle cable path will be routed around them, generating a reaction force through the sphere centre. The principle is illustrated in Figure 14. Although the effect only approximates the force vectors in some scenarios, such as where the cable wraps around the arm – which is not a sphere – the improvement is



**Figure 14. Design of virtual spherical pulleys for muscle wrapping**

(The author thanks and acknowledges Dr. David Devereux for his assistance in implementation of this component)

sufficient to generate realistic behaviour in most configurations of the body parts. Nevertheless, a logical improvement would be to extend the approach in future to cylindrical pulleys.

Models of the motors with their associated gearbox and spindle were reverse-engineered by measuring their response to a range of conditions and using multiple regression analysis to accurately parameterise typical engineering mathematical models of these components. This results in the transformation of an input voltage to a output spindle torque (see Wittmeier, Jäntschi, et al. 2011 for mathematical details). Applying Euler integration to the motor equations, the resulting net angular velocity of the spindle dictates the shortening or lengthening of kiteline in that timestep. The effects of spindle friction or slippage on the line, or changes in effective radius as the line wraps the spindle, are all currently neglected.

### 3.6.5 Statistics of complete actuated model

In total 36 active motors and associated muscle cables have been simultaneously simulated. The neck and head muscles have remained as purely passive elastic cables to date; in the robot itself their purpose is to manipulate the head for gaze direction, and they do not participate substantially in gross motor movements, but instead are simply pre-tensioned to maintain the head stability in an elastic structure. The remaining motors are employed in active control of movement and stabilising posture. The location and attachment points of these active muscles are detailed in Figure 8. In the arm and controlling the elbow joint there are the Biceps, Triceps and Brachialis. In the upper arm, torso and scapulae, controlling the shoulder joint there are the Posterior/Lateral/Anterior Deltoids which wrap the upper arm. The Infraspinatus, the Supraspinatus and the Teres Minor all wrap the shoulder ball joint. The Trapezius, Pectoralis and Latissimus Dorsi also affect the shoulder and upper arm. In the torso and back controlling the spine and posture there are the Linea Semilunaris, Quadratus Lumborum (i) and (ii), Serratus Posterior, Ilio-Costa Lumborum and the Lower Trapezius.

## 3.7 Validating the model

This section covers the steps taken from completing the passive model file with added actuation definition to obtaining a version of the model file where the model is loaded and the structure settles to a state ready to commence motor operations for training a reaching controller, according to a two-fold criteria. Firstly, that the robot model should indefinitely stand upright under its own supportive musculature. Secondly, that each motor+muscle combination has been demonstrated to be responsive to simulated voltage input and that the actuated bodies respond stably and at least subjectively appropriately within a voltage range spanning approximately a single order of magnitude.

### 3.7.1 Platform

The model was run under the ECCEOS framework (Jäntschi et al. 2010; Wittmeier, Jantsch, et al. 2011) – a C++ distributed controller developed for the ECCERobot – and employing the Coin 3D library ([www.coin3d.org](http://www.coin3d.org)) for visual rendering. (Note that all images and videos of the model in action are taken from the simulation framework developed for the ECCERobot project).

### **3.7.2 Addressing spinal issues**

It was found initially that the spine constraints, as implemented by the Bullet engine, were unable to hold the weight of the modelled robot without giving way. The model also suffered a number of further instabilities that could be resolved by employing a very short timestep ( $< 1\text{ms}$ ), but this meant that the model could run at only one quarter of real time at best, even when run on a very powerful workstation. As GPU acceleration remains, at time of writing, outside the capabilities of Bullet (in spite of its imminent addition being promised for three years) we therefore sought to streamline and adjust the model to obtain stability with useable performance.

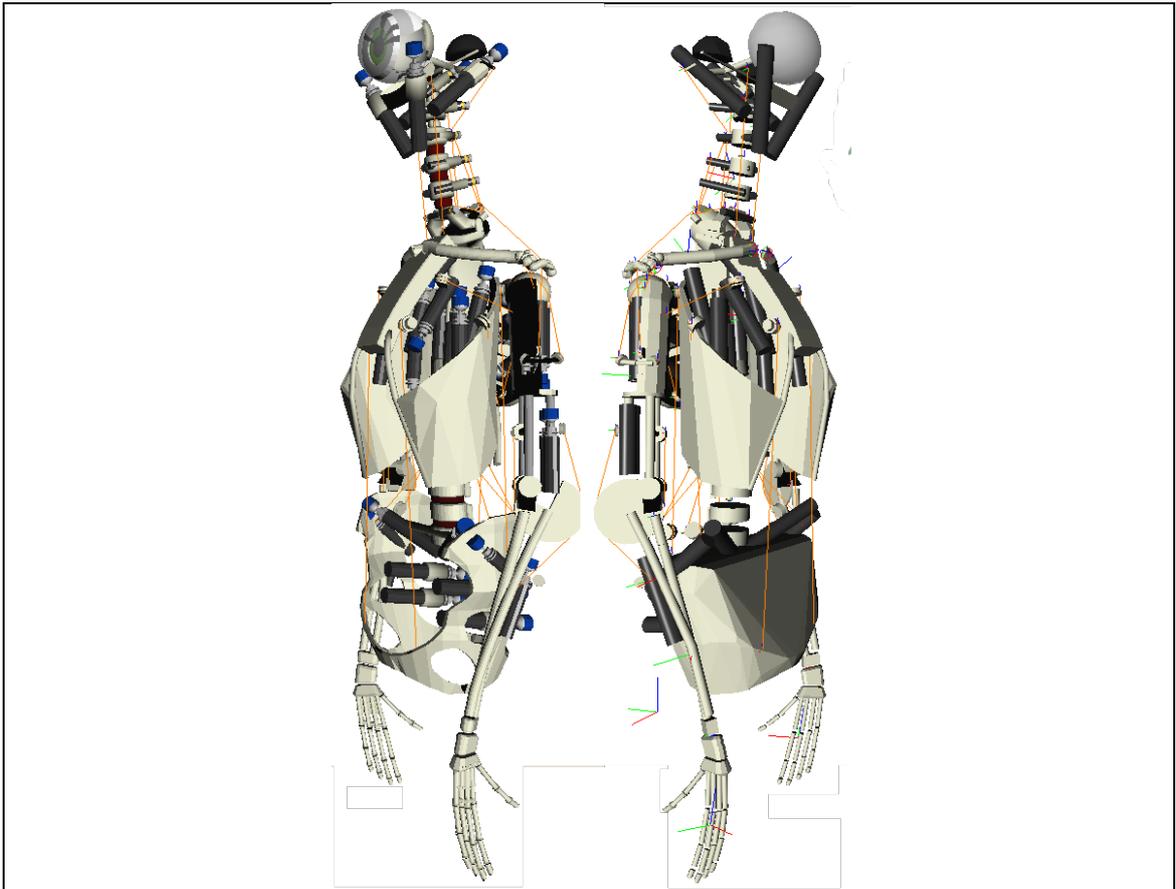
By reinforcing the inter-vertebral structure, tuning the joint friction and reducing the number of constraints in the model (such as finger joints) it was possible to raise the physics timestep to 3ms without losing stability or collapsing. This allows the model to run in real time on the workstation.

### **3.7.3 Achieving a standing state**

By pre-tensioning the back and side muscles the model robot can be made, upon loading, to settle in a stable upright stance without toppling (Figure 15). This reflects accurately both the ECCERobot and humans themselves, neither of which can remain upright without significantly tensioning the back muscles.

The muscles are pre-tensioned by altering the relevant muscle definitions in the model file to specify a starting kinteline length shorter than that from the motor spindle to the further attachment point distance (as measured between the body positions specified in the file). The tension settings were hand tuned by a iterative process of gradually raising from zero the tension of all the torso and back muscles and applying small corrections to the tensions as the model tilted and fell forward or back. To restrict the amount of tension applied the muscle tensioning was halted as soon as the model stood continuously upright albeit with a slight swaying.

It is interesting to note that the final set of tensions selected for upright standing are not inconsiderable, this is not perhaps surprising since the muscles of the back in humans are some of the largest and most powerful in the body.



**Figure 15. Stable standing under tensioned muscles**

The Visual Model (left) is formed of detailed custom meshes. For performance, the simplified Collision Shape Model (right) is formed almost exclusively of primitives

### 3.7.4 Testing responses to muscle control signals

With the model standing in its settled upright position. A simple ramp waveform was applied as a simulated voltage input to each motor in turn and the response observed and the levels noted where a clear responding movement began. It had been intended to include a normalisation parameter with each motor to scale the response to occur for the same order of magnitude across all motors. However, it was found that this was unnecessary in the event and that combining no more than two cooperative muscles was sufficient to achieve this in all cases.

## 3.8 Control research undertaken using the physics model

The completion of a stable complete physics-based model of the ECCERobot torso provides a real time, convenient and realistic platform for trialling control approaches,

including extended periods of offline learning or searching. It also potentially provides an important component in a predictive model based controller architecture, implementing features such as delay compensation and Kalman filtered proprioception, this is detailed in Chapter 6. However, research with the model has predominantly focused on the learning of reaching control using a bio-inspired approach of muscle synergies emerging under a reinforcement learning regime. This will be presented in detail in Chapters 4 and 5.

### **3.9 Conclusion**

We have shown that it is possible to construct a stable and realistic model of a complete anthropomorphic robot using a standard physics engine with some custom extensions. The result is a muscle actuated model structure that can be usefully and realistically employed to research, through real time simulation, control characteristics and issues with this family of unusual robots. However, much work remains to be completed before the model can claim to be an accurate rendition of a specific target robot. In particular, it appears very challenging matching the imprecise behaviour of Bullet constraints against what can be very complex real joints, often constructed using elastic tendons to constrain hand-moulded polymorph shapes. Indeed it is debatable whether high model accuracy can ever be achieved for such a complex ad-hoc structure and a more realistic goal may be to use the model to develop machine learning approaches that can then be equally applied to a robot with similar morphology and dynamics; either begun again from scratch or as a continuation of preliminary model-based learning. This is undoubtedly the approach taken by biology where species-specific but adaptive mechanisms passed down genetically will shape themselves around the unique morphology of a particular individual. The work on reinforcement learning of motor synergies described here is one example of such an approach. However, another potential avenue is to adaptively tune a predictive internal model instead of - or in addition to - its control.

## **Chapter 4 :**

# **Controlled Reaching Exploiting Motor Synergies**

## **Emergent Under Reinforcement Learning**

### **Part I: Algorithm Development**

---

#### **4.1 Introduction**

We describe the development and testing of a bio-inspired muscle-based approach to controlling the compliant physics-based model of the ECCERobot (developed in Chapter 3) to perform reaching tasks. By dint of the model's compliant nature, elastic muscles and numerous degrees of freedom, based closely on the robot itself, the task of control of the model was considered similar in difficulty and interest to controlling the robot itself.

In the *Background* (Chapter 2) we considered a number of avenues for locating a promising and novel approach to controlling the anthropometric ECCERobot, which we planned to trial on our extensive model of the ECCERobot (Chapter 3). We formed an initial general conclusion that learning-based, bio-inspired approaches were of most interest, in particular those incorporating means of exploiting amenable natural dynamics and compliance through action discovery rather than prescriptive trajectory planning. We therefore reviewed in detail the evidence for the success of some fully or partially bio-inspired approaches to control. In particular, we pointed to recent strong evidence from biological studies suggesting that effective control of seemingly highly complex structures such as the bodies of frogs, cats or humans is, in fact, achieved largely through a blend of amenable, evolved natural dynamics - *morphological computation* - with simple fixed pattern activations of muscle groups - *muscle synergies* - combined in simple weighted proportions.

In this chapter and the next we employ this approach to derive and test a learning-based controller design. It is a relatively novel approach, employing simple RL techniques to trigger the emergence of combinable muscle co-activation patterns to reach with a hand to any randomly presented target object location. Furthermore, by

stipulating the repetition of reward-based trials under imposed signal dependent noise we also test theoretical relationships between reliability under noise, optimal control theory and reinforcement learning to encourage the emergence of smoother, more naturalistic movement.

It should be stressed again at this point that we are not seeking here to advance the science of RL algorithms per se, such research is already conducted elsewhere. Instead, we seek to test the theory arising from the biological evidence that a muscle co-activation approach for a biomimetic structure (i.e. driving combinations of distinct fixed-weighting pattern synergies with shared *simple* signals) potentially allows relatively elementary search and learning techniques to be effective. Of these techniques, we choose to trial elementary RL for our approach, as it affords the following advantages. Firstly, every RL trial performed incrementally advances the learning, this efficiency is particularly relevant for robot learning where high numbers of repeated trials are costly in terms of time and wear and tear. Secondly, its bio-inspired nature whereby RL-like mechanisms for motor control of the body are well indicated in the CNS through the agent of dopamine. Finally, its “action discovery” focus, exploiting amenable natural dynamics and morphological computation potential of the full body structure, with the goal of maximising overall reward rather than following pre-planned, tightly controlled, trajectories in state space.

## 4.2 Overview

We first outline, at a high level, the principles and iterative process employed for learning control of the modelled robot. Subsequent sections will then detail in turn the implementation of each part of that process.

Figure 16 shows a simple standard reinforcement learning cycle for acquiring maximum reward over time through a iterative series of trials and policy improvements (Sutton & Barto 1998). The *policy* selects what *action* to take given the presented *problem state* using an estimation function that is iteratively improved by accumulated reward data acquired over an extended series of randomised trials (Monte Carlo approach). By issuing commensurate reward the outcome over time should be to optimise that performance.

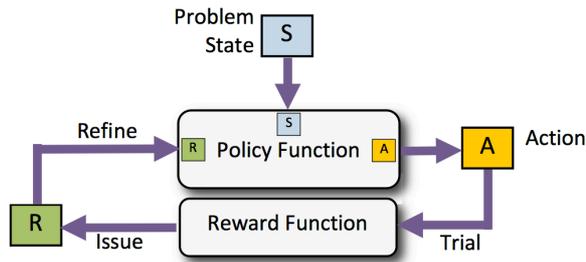


Figure 16. Standard Reinforcement Learning Cycle

As discussed in the conclusion of the *Background* chapter, we focus on the potential for control of biomimetic structures, such as the ECCERobot, through the use of simply sustained co-activations of muscles, combined in simple weightings and driven by a shared activation signal. Both the muscle co-activation pattern and the form of the signal are selected by a policy function driving from the *problem state*, which comprises the set of environmental and robot state variables intended to describe the control problem sufficient for its solution; for example, the relative position and posture of the robot with respect to a target object to be reached.

We choose this deliberately simple approach in order to test to what extent a realistic control task can be addressed by the specific means of locating an effective set of cooperating muscle co-activation patterns acting on amenable natural dynamics of the biomimetic structure in order to minimise the introduction of complexity to the controller.

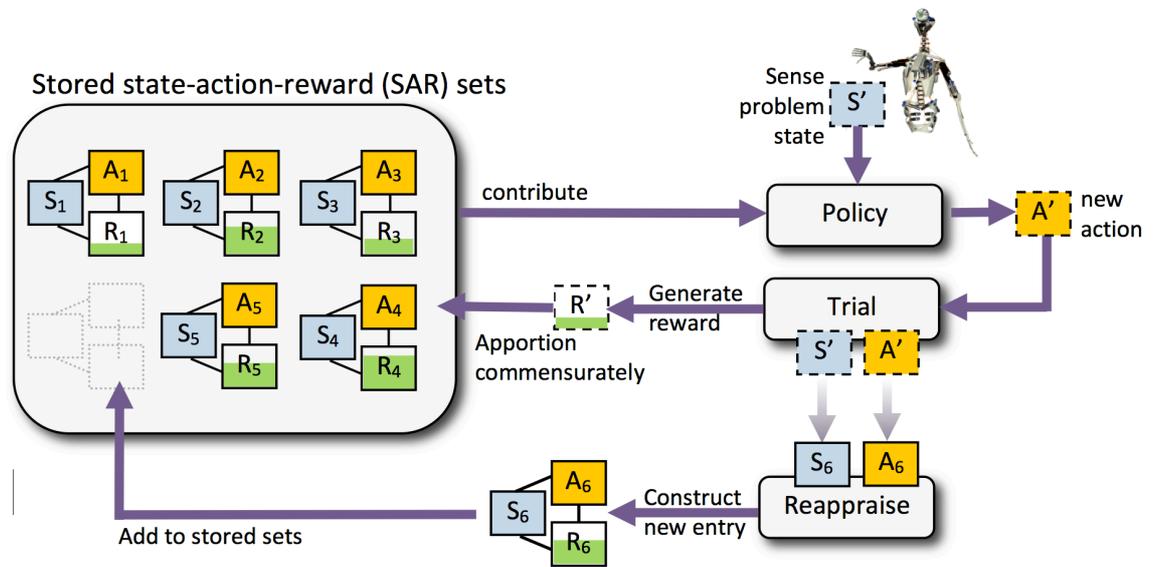
The task of our *policy* is therefore be to generate, per-trial, a single net *action* intended to address the problem state, triggering a sustained movement lasting a certain period of time, accumulating as much reward as possible as it does so, in an amount governed by a *reward function*. It should be stressed that, in order to test fully what can be achieved with the appropriate sustained muscle co-activation pattern acting in association with amenable natural dynamics, we do not look to replan actions continually at a high frequency as the state changes. Nor do we seek to apply feedback correction or muscle reflex behaviour to improve a poorly planned movement. These mechanisms may be incorporated in a later, more comprehensive controller, but would serve as obfuscation of the results if employed from the start. Instead a best new single open-loop action is estimated by the policy directly from the problem state, based on

the evidence of performance of past actions. The motor co-activations within the new action should combine with the natural dynamics to generate a new movement.

The growing set of information from completed trials that is retained for the policy to draw upon is structured according to standard RL procedure. Namely, the pairing of problem state presented and action selected (the *state-action pair*) are retained alongside the reward accrued in the trialling of the action, forming a stored State-Action-Reward (SAR) combination (Sutton & Barto 1998). Note that in robot control scenarios we are not presented with a discrete set of problem states for which we must choose between a limited set of discrete actions (as in board game, for example). Instead, the problem state space is large and continuous (and may be high dimensional) and there will be no previous action that will have addressed a given random sample point (in problem space) presented for trial. The policy must therefore estimate a new action drawing upon both the sampled problem state and the set of past (now stored) state-actions. It achieves this using a function driving from both the proximity of the sampled state to the stored states and also the past success of the stored actions (as judged by the amount of reward they have accrued).

In this case, since an action essentially comprise signal-driven, muscle co-activation patterns, our overarching aim (as discussed) is thus to locate a limited set of these patterns that are effective - specifically in linear weighted combination - in addressing a sufficiently large region of problem state space. Therefore, although a new state-action is created, trialled and eventually stored, those SARs that *contributed* to its construction are also commensurately rewarded according to both the trial outcome (reward obtained) and importantly, the size of their contribution. The approach is thus very close to the established RL technique of *eligibility-traces* (Sutton & Barto 1998) which is often used to commensurately reward earlier actions in a temporal sequence that lead to a later reward, resulting in their preferential selection in later trials. Figure 17 offers a figurative illustration of the proposed learning mechanism.

Once the contributors have been rewarded, the new action must be stored away as a SAR. As it was created through weighted combination alone, then, to encourage exploration through the influence of new SARs, the new action is first mutated by a small degree of of exploratory parameter creep (Gaussian-based, s.d. 5%).



**Figure 17: Action generation, trial and storage**

Presented with a newly sensed problem state  $S'$ , the policy constructs the  $n^{\text{th}}$  new action  $A_n$  from a combination of previous state-actions, weighted according to their proximity in state-space and past success as a reliable and effective contributor to new actions.

The new action  $A_n$  is trialed against the state  $S'$  and a reward  $R'$  obtained is commensurately apportioned between contributors. To encourage exploration through the influence of new actions,  $A_n$  is then reappraised by estimating the most suitable problem state  $S_n$  to pair it with. It is then trialed against  $S_n$ , obtaining reward  $R_n$ . The new SAR set  $(S_n, A_n, R_n)$  is added to the stored actions.

The mutated action is now reappraised by estimating the most suitable problem state to pair it with. For example, in a reaching task the original trial may reveal that the action is actually most effective at reaching to a different location. It is therefore re-trialed against this revised criteria, obtaining a correspondingly larger reward. Only now is the resultant new SAR added to the stored set.

Finally, limiting the number of stored state-actions (by pruning away the least valuable) generates a competition to be retained according to ability to act effectively as a weighted contributor to new actions. This is intended to produce, over time, a tuned policy function able to address new problem states using an underlying key set of effective muscle activation data. However, there are clearly important balances to be achieved, in the weighting and reward functions certainly, but critically between new and established actions to achieve an effective exploration-exploitation trade-off. The implementation of these are detailed in subsequent subsections.

To summarise therefore, the learning seeks to tune the policy function through a search in action parameter space, explicitly favouring the emergence and dominance of actions (driven co-activation patterns) that prove to be reliable building blocks for effective movements when used in weighted combination with other actions. For this

search to succeed we must limit the action parameter dimensionality sufficiently while retaining the flexibility to generate a range of effective movements to address the problem state space. The action parameterisation to describe both muscle co-activations patterns and temporal driving signals must therefore be designed with care (see section 4.3 below).

Most importantly, by pruning away the least valuable SARs, comprising both poor performers and poor contributors to new actions, we look to encourage the emergence of a limited distinct set of identifiable synergies (weighting patterns) than have been tuned to act effectively in concert by simple linear combination. We may then claim to have generated a form of linearizing layer such as that originally proposed in the Background (section 2.5.4), whilst the necessary reduction in dimensionality for RL-based search to succeed will have been achieved by defining actions as minimal parameterisations of sustained muscle co-activations.

It is important to note again the extent to which we are simplifying the classical reinforcement learning approach to a control task, which attempts to handle the generic case that where individual signal level on each motor can be set at each sampling interval, the robot and environment forming a new problem state at every point. Instead, we test whether, via muscle synergy combinations and natural biodynamics we can deal, not in micro, but in macro movements, covering a timescale in seconds rather than tens of milliseconds and defined by relatively very few parameters. This clearly greatly eases the learning process, but assumes however that these macro movements, when combined with exploiting the natural dynamics of the structure, have the flexibility to solve the set problems, however, the reviewed biological evidence around muscle synergies suggests they do.

### **4.3 Problem State**

In general terms the problem state  $S$  comprises a vector of both environmental and robot state variables intended to describe the control problem sufficient for its satisfactory solution; for example, the relative position and posture of the robot with respect to a target object to be reached. The minimum data set may be obtained through trial-and-error or more analytic approaches, for example, by considering

causality or mutual information measures between member variables and the endpoint of a reaching movement.

To compare two problem states we require a *proximity function*  $P$  providing a positive scalar output  $p$  based on the vector difference of states, i.e.  $p = P(S_2 - S_1)$ . The particular problem states and proximity functions used are experiment-specific and will be therefore not be defined further in this algorithm but later in the relevant method sections.

## 4.4 Actions

As discussed above, for this controller an *action* will comprise essentially a motor co-activation pattern driven by a single shared signal. Our overarching aim is to locate and store a best set of past actions that, in linear weighted combination, can estimate a new action that effectively addresses a new presented problem state. The RL-based strategy comprises a reward-led search in action parameter space, explicitly favouring the emergence and dominance of a set of stored state-action-rewards (SARs) that prove to be consistently effective contributors to a new action. For this search to succeed we must therefore limit the dimensionality of the action parameter space sufficiently while retaining the flexibility to generate a range of effective movements to address the problem state space.

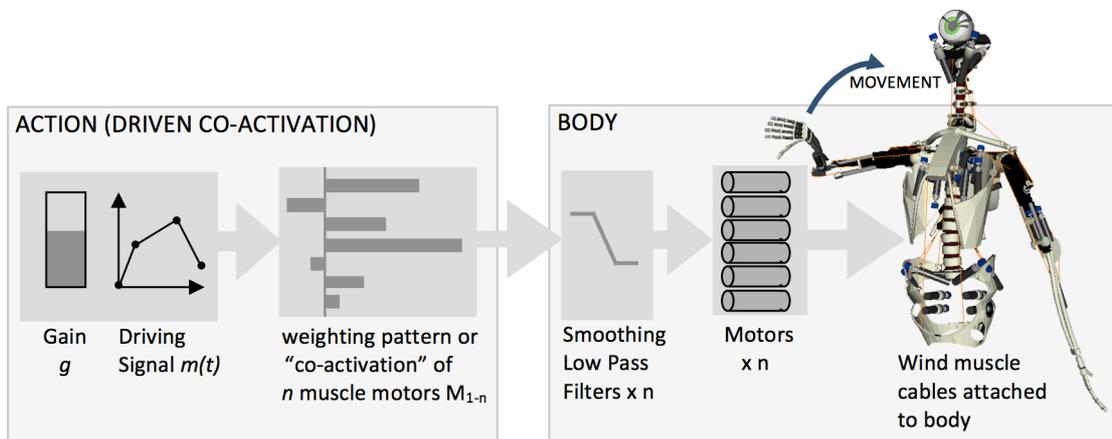
### 4.4.1 Parameterising an action as a signal-driven muscle co-activation

We define a movement as the body's response to a simple co-activation of  $n$  muscle motors configured in a specific weighting pattern parameterised as  $w_{1-n}$  weight values where  $[-1 < w < 1]$ . A negative weighting implies that the motor is driven in reverse to unwind its muscle cable, for example, this might cause a raised arm to be lowered. This weighted set of muscle motors is activated, as a single unit, by a parameterized driving signal  $m(t)$  that assumes the resulting waveform shape for a specified duration  $T$ . This concept of a driven co-activation generating movement is illustrated in Figure 18.

In order to avoid pre-empting a solution we provide the learning with significant flexibility in defining the shape of the driving signal  $m(t)$ . We parameterise with a duration  $T$  and a simple positive gain  $g$ , plus the position of 4 waypoints (see Figure 19). We choose the number of waypoints available as a minimum that can still indicate

a useful range of waveforms, from a single level or rising ramp to a non-linear curve upwards or downwards. It also makes possible the use of a period of zero level at the start or end, allowing co-activations to be potentially shifted in phase with respect to each other. Each of the ( $k=4$ ) waypoints is parameterised as a voltage level  $[-1 < v_k < 1]$  applied to the motor along with a time, held as a relative fraction  $[0 < t_k < 1]$  of the signal duration  $T$ . The  $k$ th waypoint is thus set as the point  $[gv_k, t_k T]$ .

Finally, to avoid the unwanted high frequency artefacts inherent in the raw waypoint-to-waypoint form of the driving signal we employ a digital low pass filtering function (LPF) to smooth out discontinuities before a final voltage signal reaches each motor.

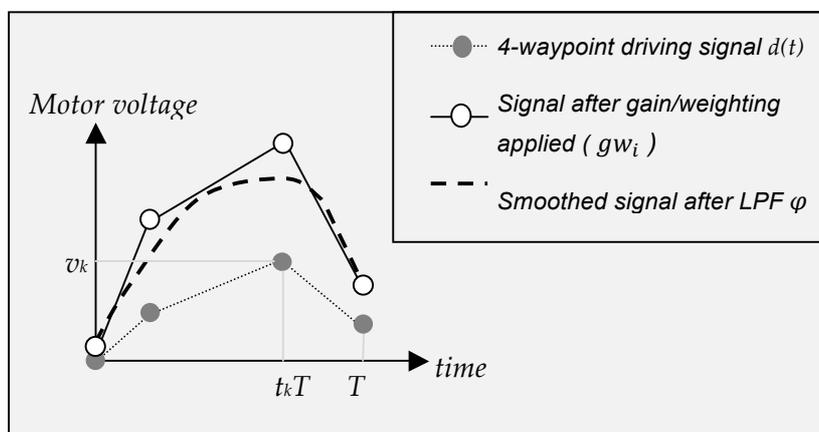


**Figure 18: Anatomy of an Action; Driving a co-activation pattern of muscle motors to cause a movement by the body**

A *co-activation* of muscle motors  $M_{1-n}$  comprises a weighting pattern  $w_{1-n}$  where  $[-1 < w < +1]$ . Note that a negative weighting implies that the motor is driven in reverse to unwind its muscle cable, for example, this may cause a raised arm to be lowered.

The co-activation is driven as a unit by a signal  $m(t)$  modulated by a simple positive gain  $g$ .

The  $n \times$  outputs from the co-activation are each passed through a low pass filter to smooth transitions and discontinuities before reaching each of  $n \times$  motors as a voltage input signal which drives it to wind/unwind its assigned muscle cable on the robot body.



**Figure 19: Parameterised driving signal used to control waveform of the motor input voltage signal**

The final voltage value  $v_i(t)$  arriving at time  $t$  at the  $i$ th muscle motor is given by:-

$$v_i(t) = \varphi (gw_i m(t))$$

where  $m(t)$  is the raw waypoint-to-waypoint form of the driving signal and the LPF function  $y = \varphi()$  is defined as:-

$$y_j = y_{j-1} + \alpha (x_j - y_{j-1})$$

where  $y_j$  and  $x_j$  are respectively the filter output and input on the  $j$ th timestep of  $\Delta t$  duration,  $\alpha = \frac{\Delta t}{\tau + \Delta t}$  and the time constant  $\tau = \frac{1}{2\pi f}$  where  $f$  is the filter cut-off frequency in Hz.

## 4.5 Policy

The policy function generates a new action based upon the problem state and SAR data stored with previous actions. In general, the  $n$ th generated action  $A_n$  is described by

$$A_n = \xi(S_n, A_{n-1}, \dots, A_1)$$

Where  $\xi$  is the policy function constructing an best estimate action  $A_n$ , driving from the  $n$ th problem state  $S_n$  generated plus the data attached to the  $n-1$  previous actions. As actions comprise muscle co-activations and our goal is specifically to locate co-activations that can act in concert with simple linear weightings, the policy constructs  $A_n$  from the linear weighted sum of the stored actions, i.e.

$$A_n = \sum_{i=1}^{n-1} \omega_i A_i$$

where the weighting  $\omega_i$  placed on the  $i$ th stored action  $A_i$  is given by:

$$\omega_i = \psi(p_i Q_i)$$

where  $Q_i$  is the *value* of the action  $A_i$  and  $p_i$  is the scalar proximity of the states  $S_n$  (new) and  $S_i$  (stored) given by the chosen proximity function  $P$  (see section 4.3).

The function  $\psi$  is a simple linear normalizing function that rescales all the  $p_i Q_i$  values proportionally between 0 and 1, whilst always summing to 1. For this investigation, we favour this linear weighting over similar non-linear tuneable functions such as *softmax* (Sutton & Barto 1998), as we are specifically seeking (as discussed) to learn

synergies that support an essentially linear weighting relationship. Thus:-

$$\psi(x_i) = \frac{x_i}{\sum_{k=0}^{n-1} x_k}$$

#### 4.5.1 Action value

The action value  $Q_i$  is defined as the average reward over time awarded to the action  $A_i$  from its contributions to creating new actions. We use action value in preference to the accrued reward in order to avoid the early favouring of actions which have been over-used due to fortuitous proximity to the random sequence of problem states. Action value also over-weights new actions, a desirable feature to encourage the immediate trialling of these new arrivals as contributors.

$Q$  is calculated as  $\frac{\text{reward}}{\text{contribution}}$  i.e. the sum, over all learning iterations, of the reward awarded to the stored SAR divided by the sum of the contribution weightings assigned to this SAR, thus the value of a stored action after  $n$  learning iterations is given by:

$$Q_n = \frac{\sum_{j=0}^{n-1} R_j}{\sum_{j=0}^{n-1} \omega_{ji}}$$

#### 4.5.2 Exploiting continuous state space to create noise driven exploration

For biological motor control no two problem states are ever identical, but simply more or less similar. As our problem state space is also continuous we can also exploit this form of “noise” driven exploration through high resolution random sampling (double precision real numbers). This effect alone will cause the state proximity function  $P$  to generate, from stored SARs, weighted contributions that vary randomly without adding any further artificial probability based exploration. Furthermore, as more actions are added and become more effective contributors, the influence of this “noise” naturally diminishes and become more exploitation than exploration. This can be understood metaphorically by considering new actions as “educated guesses” based on combined past experience. During early learning, with little experience, new actions formed by combination will be highly exploratory, rather more guess than education. Later, after considerable “education” (learning), a new action targeted at a random location will be far more accurate because of the available state proximity and effectiveness of contributing SARs.

However, whilst this approach can reduce the need for artificial techniques to encourage exploration - such as using the weighting  $\omega_i$  as the *probability* for selecting

an action (Sutton & Barto 1998) - if new stored actions are generated solely by weighted combination of past ones, then the parameter set used is by definition restricted to the outer limits of values used in the past. While this may generate widely exploratory movements at early stages of learning there is a risk of convergence to non-optimal regions of parameter space. Before storing a new action we therefore consider triggering exploration through adding an element of purely random parameter variation (see section 4.7.2).

### 4.5.3 Creating a new action from weighted combination of stored actions

Since the combination weightings sum to 1 they can be applied individually to each parameter of the stored actions to generate a new parameter value. For example, the new gain  $g_n$  parameter that will be applied to the new driving signal is calculated as:

$$g_n = \sum_{i=0}^{n-1} \omega_i g_i$$

The same formula is applied to obtain the other parameters of the new action, i.e. signal waypoints, duration, as well as the individual motor weights within the synergy patterns.

## 4.6 Trial and Reward

### 4.6.1 Reward Function

The *reward function* controls the final amount of reward issued at the end, and potentially during, the course of a trial. The system state(s) or events considered rewarding are specific to the control problem under consideration, thus a grasping task might generate reward for lifting a target object. The design of the reward function, as with the fitness function of evolutionary methods, is often critical to the success of the learning, particularly in higher dimensional spaces where issues such as local maxima can often cause learning to stall. There is generally a trade-off between rewarding only a higher goal (e.g. “obtain food” ) and incorporating additional staging rewards that may constrain solution freedom but act to smooth a more jagged reward “landscape” by providing clearer “hill-climbing” routes for incremental learning to follow (e.g. “move nearer to food”). The precise reward function employed by the algorithm will be specific to the experiment and therefore detailed in the relevant method section.

#### 4.6.2 Trial repetition and signal dependent noise to leverage optimal control

As discussed in the Background chapter, observed smooth, efficient human movement suggest the presence of optimal control mechanisms, acting to minimise a cost function, resulting in the typical bell-curve velocity profiles of eye saccades or reaching (e.g. Collewyn et al. 1988).

Although the underlying cost function employed in motor control was believed to be minimisation of jerk (Suzuki et al. 1996; Breteler et al. 2002) this theory has since been superseded by a cost function minimising endpoint variance in the presence of amplitude-related motor neuron noise (Harris & Wolpert 1998; Miyamoto et al. 2004). In other words, selected movements are those most reliable over repetition, a very clear benefit to the subject. In unreliable or noisy systems RL will, over repetitions, inherently favour the most reliable solutions as they will accrue the most reward (Wolpert et al. 2001). We therefore propose to draw upon this optimal control principle to encourage the emergence of smoother, more naturalistic movement by incorporating signal dependent noise and trial repetition.

To this end, we artificially add signal dependent noise to the motor signal, applying a linear relationship between the standard deviation of the motor signal and its amplitude. Note that, although measurements have shown individual spiking units to exhibit a log-log relationship (with a slope of close to 0.5) individual variation in threshold level means that for a population of spiking units the slope of the log-log in fact approaches 1.0, i.e. a linear relationship (Jones et al. 2002). Indeed, it has been shown that for a square root relationship (slope=0.5) optimal control converges to a bang-bang strategy (full on or full off), whereas the linear relationship optimally predicts a smoothly varying control signal (Jones et al. 2002; Miyamoto et al. 2004), as is observed in nature (Collewyn et al. 1988).

To create our noisy signal therefore, on each timestep, artificial noise  $n_g$  is generated and added to the motor voltage just before reaching the motor itself. Following recommendations from motor neuron studies (Jones et al. 2002; Hamilton et al. 2004), Gaussian distributed noise is employed (i.e. drawn from a normal distribution with mean=0 and s.d.=1) as this has been shown to match motor neurons firing at above 5 pulses per second (pps).

The final voltage  $v_{noisy}$  reaching the motors is given by:

$$v_{noisy} = (1 + kn_g)v_{clean}$$

where  $[0 < k < 1]$  is a tuning parameter setting the noise level by controlling the variance in the Gaussian noise. In these experiments we set  $k=0.2$ , corresponding to the lower values observed in nature (Hamilton et al. 2004).

We then repeat the trial of an action for the same problem state  $N$  times, thus issuing  $N$  sets of reward. Over time this should favour the emergence of actions that are both accurate and also reliable. Note that the experimental setting of the value of  $N$  to a practical value will be discussed in section 4.9.4 covering early trialling of the controller.

## 4.7 Policy update

The policy function creates new actions to address a random sampling of problem states and is incrementally improved using two mechanisms based on information provided by trials of earlier actions, in the form of reward. The first mechanism is to update, in the light of the new information, the *value* of the stored SAR sets which contributed data to the formation of a trialled new action. The second mechanism is to extend the set of stored SAR sets with an  $n$ th new entry comprising a problem state, an action and an initial reward. The aim is, at an early stage, for the policy to explore substantially different actions as contributors to new actions while, at a late stage, to continually refine actions.

### 4.7.1 Updating values of stored SAR contributors

A new action  $A'$  is first trialled using the randomly generated problem state  $S'$  and commensurate reward  $r$  issued as per the reward function (see section 4.6.1 above). Trials of  $A'$  to solve  $S'$  are repeated  $N$  times in order to favour – i.e. generate more reward for – actions more reliable under signal dependent noise (see section 4.6.2 above). The average reward  $\bar{r}$  accrued per trial repetition is now divided proportionally among the stored SARs according to their contribution to the new action, as specified by the weighting  $\omega$  that was assigned by the policy (see section 4.5.3).

Thus, the total reward  $R_j$  attached to the  $j$ th stored SAR is updated as:

$$R_j \rightarrow R_j + \omega_j \bar{r}$$

However, the policy requires that we update the SAR *value*  $Q = \frac{\text{reward}}{\text{contribution}} = \frac{R_j}{C_j}$

We must also therefore update the total contribution  $C_j$  made by the SAR:

$$C_j \rightarrow C_j + \omega_j$$

#### 4.7.2 Constructing and reappraising new action based on best problem state

Once contributors have been rewarded according to the presented problem state we consider the task of creating a new SAR stored entry. Although the simplest approach would be to use the action  $A'$ , the presented problem state  $S'$ , and average reward  $\bar{r}$  accrued in trialling (see Figure 17) this has some disadvantages.

##### 4.7.2.1 *Weighted combination alone may limit parameter space exploration*

The first disadvantage is that, if new stored actions are generated solely by weighted combination of past ones, then the parameter set used is by definition restricted to the outer limits of values used in the past. While this may generate widely exploratory movements at early stages of learning there is a risk of convergence to non-optimal regions of parameter space. To address this, we re-create the new action parameters from the original weightings whilst adding a small degree of Gaussian-based random variation, (referred to as *mutation* in evolutionary algorithm parlance). For example, the new gain  $g_n$  parameter is re-calculated as:

$$g_n = (1 + kn_g) \sum_{i=0}^{n-1} \omega_i g_i$$

where artificial Gaussian noise  $n_g$  is generated, then clipped to the range  $[-1 < n_g < 1]$  ) and  $k \approx 0.05$  is a tuning parameter scaling the maximum mutation effect to around 5% by controlling the variance in the Gaussian noise.

We raise the caveat however, that unless the noise level is significantly raised, this approach is likely to have relatively little effect on creating actions containing parameter values that move beyond the highest and lowest values used in the stored set. This issue may ultimately be better addressed by providing a sufficiently wide-ranging initial seeding set of actions or, alternatively, by employing extrapolation to

estimate actions to address problem states outside of the region covered by the stored set. Whilst unlikely to generate a particularly effective action this would nevertheless serve to extend the covered region.

#### **4.7.2.2 Problem state is sub-optimal for the new action**

A second disadvantage of the simplest approach to adding a new SAR is the retention of the original problem state  $S'$  generated. It is highly likely, particularly in the earlier stage of learning, that the new action  $A_n$  could generate more reward if judged against a different problem state. For example, in the early stages of learning a reaching task, a new action might be created that causes reaching to a point  $P_2$  in space, considerably to the left of its intended target,  $P_1$ . Although it may gain some reward (depending on the reward function) if the problem state had specified the target as being located at  $P_2$  it would have generated a much higher reward in trials. Creating a SAR stored entry using this revised problem state and the resulting higher reward not only forms a stronger entry but also provides a powerful exploration and action discovery element, particularly at an early stage of learning, by shifting the problem state for the new action away from that predicted by simple action combination.

To achieve this, the problem state is therefore reappraised from trial data of the movement triggered by the (reconstructed) new action, aimed at identifying a problem state that would generate more reward than the original. In practice, this will involve primarily environment state variables (e.g. reaching target location) and should converge towards the original problem state as the policy improves over multiple iterations. Once again, the particular function to calculate this "better" suited problem state is experiment-specific and will be provided in the relevant method sections. The resulting revised problem state  $S_n$  will be used in the  $n$ th stored SAR alongside  $A_n$ .

#### **4.7.3 Assigning reward to new actions**

The final task in adding a new stored SAR is to select the starting reward to assign, placing it most appropriately in relation to older entries. Although the new action has been trialled in isolation it remains untested in its ongoing role as a contributor to new plans, and may or may not prove effective as such. Ideally a new action should be overused for the short-term to test its validity and if it does not prove a good contributor over the longer term it should fall back. To achieve this, it must therefore enter with a high  $Q$  value.

We can obtain this effect relatively easily by simply treating the new action as being created by a weighted contribution  $\omega = 1.0$  and reward  $R_n$  obtained from its trial judged against its revised best-fit problem state  $S_n$ . This allows it to begin alongside the previous actions with a value (average reward)  $Q = \frac{R_n}{1.0}$ . This will comprise a relatively high entry as it was obtained against the best-fit problem state for the action allowing this action to contribute heavily in the short term to addressing new problem states that are close to its own. This means it can establish itself as a strong contributor if the new actions created prove effective. If this is not the case, then its average reward ( $Q$ ) will drop relatively more rapidly than older actions as it has undergone few contributing iterations.

#### **4.7.4 Limiting stored plans to encourage emergence of an effective dominant set**

Over the course of numerous iterations a large number of new actions are generated and stored. Apart from slowing down the selection and combination process, retaining so many actions causes reward to be thinly dispersed between numerous similar actions and prevents a clear set of dominant, effective synergy patterns from emerging. Recall that the desirable outcome is to construct, in effect, a limited “library” of SARs which comprise that set most effective at addressing any new problem state by means of weighted combination, where the weighting function drives linearly from the problem state. We therefore implement a competitive elimination process that retains only a maximum number  $N_A$  actions by removing one action on every iteration once the maximum is reached. Identifying the minimum effective  $N_A$  and any resultant effect on the retained plans’ characteristics as  $N_A$  is reduced are points of interest that will be considered during experiments. There are numerous criteria that might be applied to selecting the action for removal, but we start by implementing a very simple one, namely removal of the action with the lowest value,  $Q$ . This approach can easily be refined if necessary, for example, ensuring an even coverage is retained in problem state space by incorporating proximity between SARs into the removal criteria.

## **4.8 Complete learning algorithm**

The complete learning cycle described over the previous sections is summarised overleaf (Figure 20).

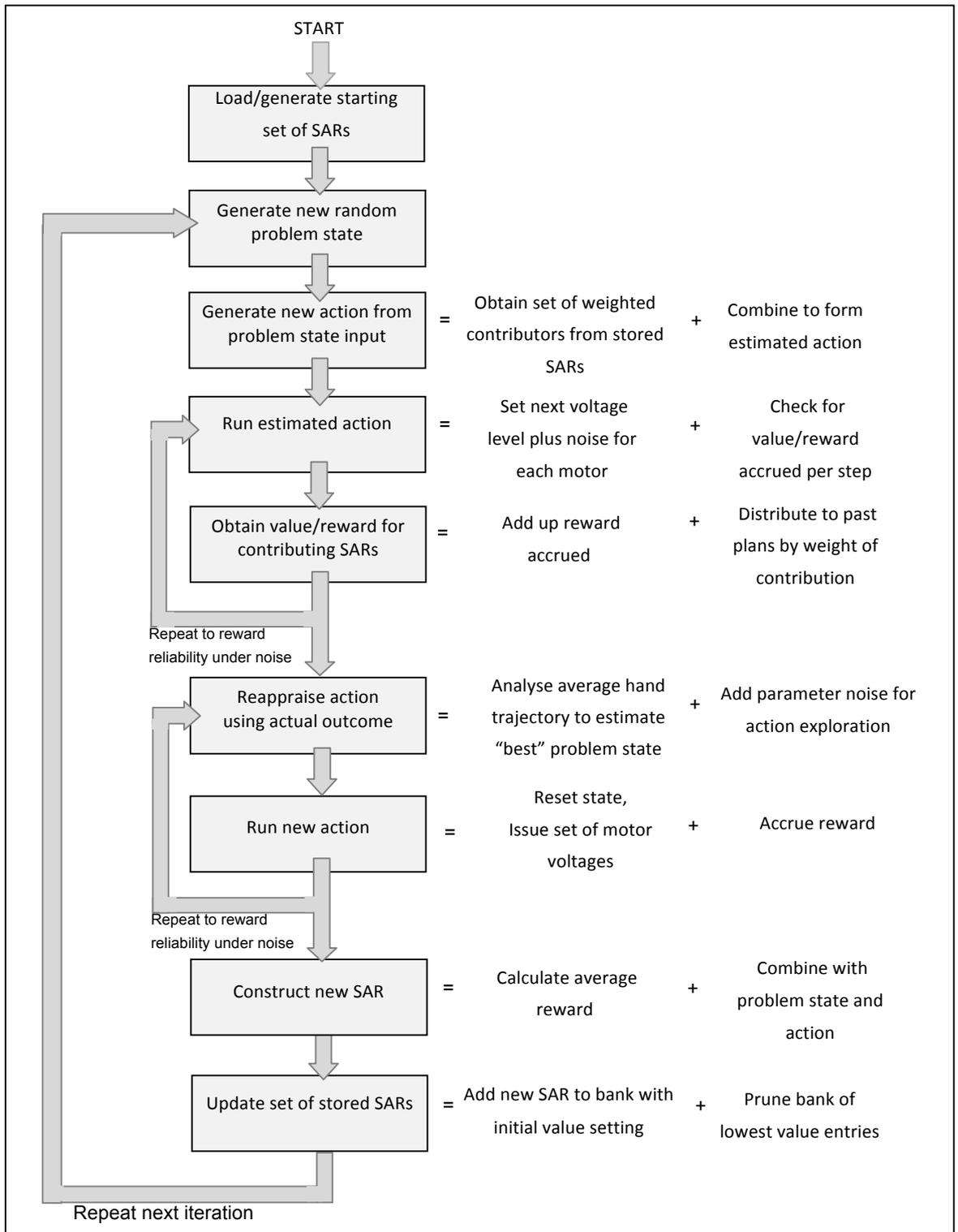


Figure 20: The final learning flow algorithm

## **4.9 Initial feasibility investigation**

### **4.9.1 Creating functional movements using action definition and parameter set**

Prior to attempting machine learning an initial feasibility investigation was conducted to test whether the structure and parameter set of an action was sufficient to obtain useable movements. It was found that functional reaching movements could be manually constructed from the parameter set of an action with reasonable ease. Whilst the final hand trajectory, or the point in space it ultimately reached, was not known in advance, once a useable action was uncovered it could be reproduced if the robot model began from a relatively constant starting “rest” state. Furthermore, adjustments to the global gain setting for the same plan, also tended to produce useable movements, as (subjectively) more or less exaggerated forms of the same movement. In general, this process often did not degenerate movement rapidly into instability or chaotic movement until the upper limits of gain were approached. Lowering the gain moved relatively quickly to a point where the arm could not be lifted at all due to insufficient force. Similarly, variations can also be easily generated by altering the duration of plans by time stretching or compressing the activation signal. As with the gain variation, this approach also tended to produce functional variations.

### **4.9.2 Testing action combination algorithm**

We tested the action combination algorithm (see above), finding that merging two similar actions usually generated a result in between its “parents” outcomes, with this outcome becoming increasingly unlikely with their dissimilarity. For example, as one would expect, the given the non-linearity of the structure, essentially horizontal movements combined with vertical ones do not produce an outcome “halfway up and across”. Nevertheless, as is generally the case with solutions to non-linearity, issues recede if we consider smaller parts of the problem at a time – in this case, when limiting combination to small differences in the “parents” used to form new actions. Meanwhile, combining dissimilar actions appears a useful driver to generating novel actions during an earlier, more exploratory phase.

### **4.9.3 Tonic muscle activation**

It was noted that during a movement it is not uncommon for some muscle cables to become slackened if they are not actively participating in the generation of the movement at the time that the distance from their attachment point to pulley is being

shortened as a result of the movement. This causes issues with subsequent actions that require participation of this muscle since driving the motor spindle has no effect until the slack has been taken up again – a very unnatural effect. To combat this, on each timestep, muscle cable lengths (calculated by the motor spindle simulation) are compared to the current physical distance between anchor/pulley points. If muscle lengths exceed the distance then a tonic +ve voltage is sent to wind the excess cable. The speed of winding is set by the voltage such as to fully wind the loose cable and eliminate the slack by the end of the next motor voltage timestep.

#### 4.9.4 Trial repetition to exploit optimality theory

As discussed, action trials are to be repeated  $N$  times to investigate whether smoothness and reliability of movement can be raised by using RL under signal dependent noise. Note that the choice of  $N$  is an issue as it impinges directly upon on the overall learning time as a large proportion is taken up by trialling due to the relatively slow model speed. However, to obtain a statistically representative spread of noise-distorted motor signals performing more repetitions is indicated.

To locate a low but usable value for  $N$  we took three sample reaching actions, each trialled 60 times under noise, and measured the endpoint standard deviation (*ESD*). Table 1 shows the standard deviation of *ESD* within groups of 5, 10 and 20 consecutive trials. We use three samples as any one action alone may be more or less susceptible to endpoint variance caused by the signal dependent noise.

The results suggests that whilst using just 5 repetitions of trials proves less consistent than 20, it remains of the same order. As this also offers a fourfold increase in learning rate we therefore commence experiments using  $N=5$ .

	<b>N=5</b>	<b>N=10</b>	<b>N=20</b>
<b>Action 1</b>	3.53cm	2.79cm	1.83cm
<b>Action 2</b>	3.37cm	2.57cm	1.96cm
<b>Action 3</b>	4.15cm	3.83cm	3.02cm

**Table 1. Stabilising statistical endpoint variation**

Three sample reaching actions, each trialled 60 times under noise. After measuring the endpoint standard deviation (*ESD*) for each we display the standard deviation of *ESD* within groups of 5, 10 and 20 consecutive trials.

Overall, we considered these findings to be sufficiently promising to proceed to a full experiment design to test the approach developed in this chapter.

## **Chapter 5 :**

### **Controlled Reaching Exploiting Motor Synergies**

### **Emergent Under Reinforcement Learning**

### **Part II: Experiments and Results**

---

#### **5.1 Overview**

In the previous chapter we derived an algorithm approach intended to generate, reward and combine muscle co-activation-based motor plans to progressively improve performance of a motor task over numerous iterations. In this chapter we test the approach, commencing with a simple reaching experiment that repeatedly offers target objects at random locations to the complete modelled ECCERobot and issues graded rewards for success in reaching to them.

Following positive results we present an analysis of the findings including the performance and characteristics of the learning algorithm and the effects of compliance on control of complex musculoskeletal structures.

By performing repeated trials under imposed signal dependent noise we also test theoretical relationships between reliability under noise, optimal control theory, reinforcement learning and minimum-jerk.

Most pertinently, we next draw upon factor analysis methods to examine the emergence or otherwise of recurring muscle activation patterns - which we label *candidate synergies* - within the set of muscle co-activation-based actions built up through learning. To investigate whether these can act as “true” synergies, we compare the performance of a revised, low-dimensional controller using these synergies explicitly as fixed units, rather than individual muscles.

Finally, we discuss a number of extensions and other future work. The most relevant of these is the potential for the transfer of this approach to control of real (i.e. not models) musculoskeletal robots such as the ECCERobot itself. We therefore propose extensions

to the learning algorithm that may sufficiently reduce the number of learning trials required to within the wear and tear limitations of the physical ECCERobot.

Other proposals address more realistic and complex problem scenarios such as starting from any dynamic state and incorporating the controller and model as a planner and state prediction modules within a general control architecture (covered in detail in Chapter 6). We also consider the potential benefit of, and scope for, creating hybrid approaches by selectively merging this approach with other established techniques.

To conclude, we present a list of potential implications arising from this work for theories of neurological motor control.

## **5.2 Learning to perform a reaching task**

### **5.2.1 Introduction**

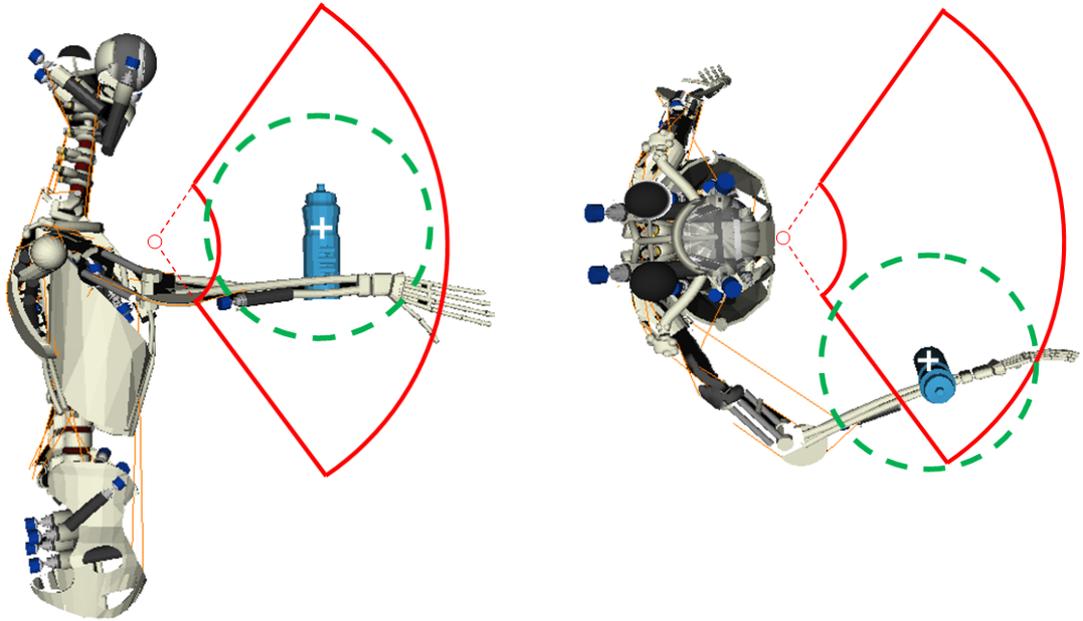
We describe an experiment that applies the approach developed in the previous chapter to the problem of controlling the physics-modelled ECCERobot to reach to a target object placed at successive random locations.

### **5.2.2 Problem State**

This experiment considered a simple control scenario with the model robot's pelvis anchored to a static immovable base. Each trial commenced with a model reset to a "ready position" such that the robot is held upright under pre-tensioned torso and back muscles, with arms at its sides. By employing this same starting position on each trial, the problem state simplifies to comprise solely the randomly generated position  $(x,y,z)$  of the target object. To avoid potential subtle control dependencies caused by precisely identical starting states we vary randomly, by up to 0.5 seconds, the timestep when an action begins to activate. This delay causes a small but useful variation in dynamic state since, upon loading, the model the robot remains upright but continues to sway slightly, with its arms noticeably swinging.

### **5.2.3 Target Object and Placement**

The target object selected for reaching was a physics-based model of an empty plastic bottle of mass 200g. This was chosen primarily to complement the trial of the Kinect-based vision system developed (Devereux et al. 2011) for the ECCERobot (see Chapter 6 for further details). For each reaching trial the bottle model is placed into the physics



**Figure 21: Side and top view of reaching experiment**

For each trial, the centre of mass of the bottle model is placed at the target location (white cross) which is generated at random per-trial within the zone denoted by the red lines. The green sphere centred on the white cross indicates the extended zone for obtaining some reward by proximity of the hand to the target.

scene in front of the robot model at a random location, but within a limiting spherical zone. The experimental setup is illustrated in Figure 21.

The bottle is balanced on a minimal static base, the intention being to support the bottle without interfering, through collision, with movements generated by the controller. Note that, if the bottle is dislodged by the robot, the base is immediately removed from the scene. This simply prevents the robot arm or hand from becoming lodged on the static base and potentially obtaining an undue amount of reward as a result.

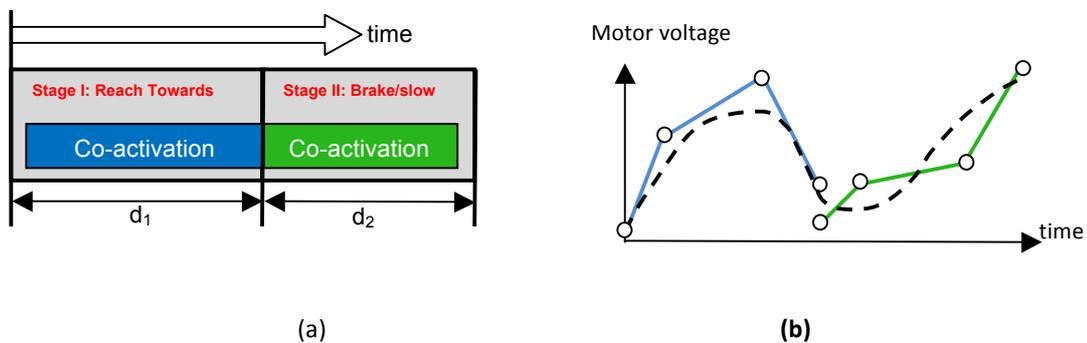
#### 5.2.4 Definition of a reaching movement

One of the overarching aims of the ECCERobot project was to demonstrate control of a simple reach and grasp of an object. For this experiment we therefore define an ideal reaching movement as reliably (i.e. with repeatability under noise) moving either hand from its (approximately constant) starting location to a target object located at a point  $(x,y,z)$  in space and slowing or stopping it there so as to potentially enable a successful grasp. As the physics model employed lacks a jointed or muscled hand (for

performance reasons - see modelling Chapter 3) we do not extend to attempting an actual grasp at this stage.

Our approach assumes that richer behaviour can be obtained by sequencing muscle co-activations in time, forming a compound action that produces a multi-stage movement. Figure 22 illustrates how this chaining of co-activations is implemented, considering the issue of the switchover point in particular.

To recap, we are seeking to test the idea that applying appropriate sustained muscle co-activation patterns can successfully comprise much of a control solution when combined with amenable natural dynamics. We therefore begin this reaching-based experiment with a simple assumption that a compound movement for reach/grasp can be achieved by only two muscle co-activation stages: the first co-activation to generate a movement of the hand towards the target and a second to slow or hold the hand on arrival. Note however that we will place no explicit stipulation on the roles of either stage, beyond designing the RL reward scheme to reward both reaching to physically touch (or strike) the target and also maintaining the hand as close as possible for as long as possible. It may therefore quite legitimately emerge that, in some subset of problem states, the second stage takes a different role, perhaps acting as a direction correction mechanism in the case of movements to target locations where a single co-activation is insufficient to generate an accurate trajectory.



**Figure 22: Anatomy of a compound action**

- (a) An example of a compound action, comprising two co-activations, for triggering a potential reaching behaviour. Each stage is the result of a separate co-activation which may be of different duration from each other.
- (b) When the action is invoked the two activations (blue and green) defined for a given motor are executed consecutively, but pass through the smoothing low pass filter before becoming a motor voltage signal (dashed trace), this acts to reduce transients from the switchover.

For the policy to estimate a new action from a weighted combination of effective past actions we combine individually the actions of each two co-activation stages by following the algorithm described previously (see section 4.5 and 4.5.3).

Note also that, in this first experiment, selection of which hand to use is not part of the learning. Instead, the nearest hand to the target position is explicitly selected on each presentation and motor signals intended for the left hand are simply issued as mirror images of those for the right. However, it is important to note that muscles from both sides of the body can form part of any given co-activation pattern. For example, leaning to the side could involve an agonist-antagonist cooperation between left and right side muscles.

### 5.2.5 Reward Function

Qualitatively, reward is issued as a result of (preferably lightly) touching or striking the target. A light touch is favoured over striking at speed since this would prove most conducive to a subsequent successful grasp of the bottle. However, early tests confirm that learning commences and progresses considerably better if the reward “landscape” is smoothed by providing a secondary reward for reaching to, at least, the vicinity of the target. We do not reward for any shape or trajectory of movement other than the indirect effect of rewarding reliability under signal noise through the use of trial repetitions. This freedom is intended to fully exploit both conducive natural dynamics and the extensive redundancy in the robot structure (there are an infinite number of actions that can reach a given target).

#### 5.2.5.1 *Strike/touch reward*

In these experiments with a non-jointed hand, no attempt is yet made to pre-shape the approach or physically grasp the bottle, and reward is issued for striking, or preferably, simply touching the bottle. A strike reward  $R_s$  is therefore issued when the strike is detected but the amount is set inversely proportional to the absolute hand speed  $v$  at that time step. Hand speeds above  $v_{fast}$  ( set to  $0.5\text{ms}^{-1}$  ) are treated uniformly as “fast”. Thus:

$$R_s = \kappa \left( 2 - \frac{v}{v_{fast}} \right) \quad [v \leq v_{fast}]$$

$$R_s = \kappa \quad [v > v_{fast}]$$

where  $\kappa$  is a scaling parameter to set the strike reward relative to the secondary zonal reward (see below). The ratio  $\frac{v}{v_{fast}}$  provides a unit-free indication of the hand speed. We set a minimum reward of  $\kappa$  for any strike and a maximum of  $2\kappa$  for a perfect touch ( $v = 0$ ).

### 5.2.5.2 Zonal reward

As discussed above, a secondary reward mechanism was found to be of considerable help to kick-start early learning actions towards the vicinity of the target. A scaled zonal reward  $R_z$  is therefore also issued for every timestep that the centroid of the reaching hand is located within the spherical reward zone surrounding the target (see dotted green zone, Figure 21). This also doubles as a mechanism to reward the hand remaining held as close as possible to the target, thus maximising the chance of a successful grasp.

A simple linear measurement was found to be effective in scaling the amount of reward issued per step, namely the proximity of the hand centroid to the target centre. However, it was more effective to limit this reward to a limited zone, rather than simply issuing suitably scaled reward at any hand location. This causes the learning to explore until this zone is located, rather than commencing hill-climbing until potentially lodged in local optima in regions far from the target. The full zonal reward  $R_z$  is therefore the sum of the incremental  $\Delta R_z$  reward issued per timestep that the hand is within the zone, given by:

$$\Delta R_z = 1 - \frac{d_t}{r} \quad [d_t \leq r]$$

$$\Delta R_z = 0 \quad [d_t > r]$$

$$R_z = \sum_{t=1}^T \Delta R_z$$

where  $d_t$  is the distance from the hand centroid to the target centroid on timestep  $t$ ,  $r$  is the radius of the spherical reward zone around the target (set to 20cm) and  $T$  is the number of timesteps that the hand remains within the reward zone.

### 5.2.5.3 Scaling of strike and zonal reward

For scaling purposes relative to strike/touch reward, zonal reward  $R_z$  is limited to a maximum value equivalent to holding the hand at the target ( $d_t \equiv 0$ ) for one second.

The value of  $\kappa$  in the calculation of the strike reward  $R_s$  is set to provide the same maximum reward.

#### 5.2.5.4 *Total reward*

The combined reward  $R$  issued for the trial is therefore given by:

$$R = R_s + R_z$$

#### 5.2.6 **Dimensionality of muscle space**

Thirty-six modelled muscle motors (18 left + 18 right) were made available to form potential muscle co-activation patterns. The location and attachment points of each muscle are detailed in Chapter 3 (Figure 7). In the arm, controlling the elbow joint are the Biceps, Triceps and Brachialis. In the upper arm, torso and scapulae, controlling the shoulder joint there are the Posterior/Lateral/Anterior Deltoids which wrap the upper arm. The Infraspinatus, the Supraspinatus and the Teres Minor all wrap the shoulder ball joint. The Trapezius, Pectoralis and Latissimus Dorsi also affect the shoulder and upper arm. In the torso and back controlling the spine and posture; the Linea Semilunaris, Quadratus Lumborum (i) and (ii), Serratus Posterior, Ilio-Costa Lumborum and the Lower Trapezius.

All of these muscles are mirrored left and right. Note that the many active muscles of the head and neck (trapezius apart) present in the physical ECCERobot are excluded from the controller as their main purpose is to stabilise the head and control gaze direction, and they are not otherwise employed in task driven movements. They are included in the physics model as they nevertheless exert some influence on the dynamics of the structure and hence a movement generated by a motor action. They are placed under a fixed preset tension to hold the head stable on the neck vertebrae but are removed from the learning in order to significantly reduce dimensionality at little cost to control.

#### 5.2.7 **Initial set of stored actions**

An important consideration is the default set of motor plans that are supplied at the beginning of learning. As discussed, in high dimensional spaces any learning approach functions significantly better as an optimiser or improver than an explorer (Schaal 1999). This principle is widely applied already, through the use of “imitation”, in the field of humanoid robot control. Termed “co-active learning”, reaching would be kick-started by the researcher holding the robot’s hand and moving it to the target. This

results in effectively focusing the learning to a region of the otherwise very large search space, where it is likely the best solutions lie (Schaal et al. 2003).

In our case, we have a model rather than a physical robot to co-act with, but we may still draw upon the principle of this proven approach to obtain an initial set of stored SARs before the main RL learning cycle is commenced. Note that, in all cases described here, each selected member of this set is tested and stored as an SAR by the same algorithm as the new actions that will be subsequently produced by the RL cycle (see section 4.7.2 and Figure 20).

A first set of ten functional actions were selected from those developed by hand during the initial feasibility testing (see section 4.9). The selection focused on spanning a range of endpoints and trajectories and ensuring that all available muscles were represented across the selection. This focus on range reflects the recognition that the algorithm places an emphasis on generating plans by averaged combination over the gradual creep introduced by random noise/mutation.

“Hand-rolling” one of these initial rough muscle activations followed a simple standard process whereby a reaching endpoint is first selected, prioritising a relatively unexplored region of target space. Next, activation of each main muscle is trialled in isolation or at most pairs to locate a simple muscle activation that brings the hand closest to the target, leveraging available amenable natural dynamics. Minor activation of other muscles is now added to adjust the trajectory closer to the target. Simplicity is prioritised over precision as these nominal endpoints will play little direct role in the final controller.

A supplementary set of 20 further actions was also generated using functional variations of the first 10 actions created by gain alterations and time stretching. A particular emphasis was placed on generating actions that moved the hand nearer the outer reaches of the target placement zone because the policy will perform better filling the gaps in the problem state space, a far easier goal for combination techniques, rather than moving away into the outer reaches, a process driven by slower noise-driven parameter creep.

### 5.2.8 Policy Function

To recap (see section 4.5.3) the policy constructs the  $n$ th new action  $A_n$  from the linear weighted sum of the existing actions, i.e.

$$A_n = \sum_{i=1}^{n-1} \omega_i A_i$$

where the weighting  $\omega_i$  placed on the  $i$ th stored action  $A_i$  is given by:

$$\omega_i = \psi(p_i Q_i)$$

where  $\psi$  is a simple linear normalizing function that rescales all the  $p_i Q_i$  values proportionally between 0 and 1, whilst always summing to 1, and  $p_i$  is the scalar measure of proximity between the new randomly selected problem state  $S'$  and the state  $S_i$  attached to the stored action  $A_i$ .

In this experiment, the problem state comprises only the target location and excludes more complex states, such as that of the robot itself. We therefore simply set  $p_i$  to be directly proportional to the absolute distance  $d_i$  in 3D space from the new target location  $[x_n, y_n, z_n]$  to the target location  $[x_i, y_i, z_i]$  attached to the  $i$ th stored action:

$$p_i = \left(1 - \frac{d_i}{d_{max}}\right)$$

where  $d_{max}$  is the maximum distance that one target location can be from another, i.e. equivalent to the diameter of the target placement sphere (see section 5.2.3 and Figure 21, red zone). This formula provides the desired linear dependency from problem state to weighting that the RL will seek to conform to by its competitive selection of those SARs that remain in store.

### 5.2.9 Creating new SAR to store

In section 4.5.3 we derived how a new SAR is created from an estimated action generated by the policy. However, the function defining a “best” problem state  $S_{BEST}$  to attach was not specified since  $S_{BEST}$  depends on the learning scenario in question. We therefore now define  $S_{BEST}$  (comprising solely a target location in this experiment) as the point  $S_{min}$  within the target zone (red zone, Figure 21) along the reaching hand’s trajectory where the hand speed reaches a minimum, obtained by interrogating the hand speed while the action  $A$  is trialled. In fact, since the action is trialled multiple times (see 4.6.2 below), we take the average of  $S_{min}$  to store along with the new action and its initial value calculated from the average reward accrued.

## 5.3 Implementation and experiment parameters

### 5.3.1 Implementation platform

The learning architecture employed was implemented in C++ as a separate module within the ECCEOS simulation framework developed by the ECCE team (for details see Wittmeier et al. 2011).

### 5.3.2 Stored SAR limit

For this experiment an initial maximum setting of  $N=100$  SARs were retained, excess SARs are removed by lowest  $Q$  value (see section 4.7.4). Note that a further investigation where the setting was dropped as low as  $N=30$  is discussed later (see section 0).

### 5.3.3 Clocking rates for the simulator and physics model

The timestep for the physics model is set at 3ms (simulated), which had been identified as providing best performance versus stability trade-off (see modelling Chapter 3). As discussed in the previous chapter, the model can run at close to real time, but no faster without GPU acceleration.

The control signals and learning algorithm do not require such fine granularity and are set to update on a timestep of approximately 100ms. Every 33 physics steps the control signals sent to motors are updated according to their underlying action parameters (driving signal + co-activation pattern). The reward function is also triggered at the same time, allowing it to interrogate the state of the model and issue any intermediate reward by comparing it with the problem state. In practice, for this simple experiment at least, this entails comparing the reaching hand position with the target location.

### 5.3.4 Learning trial duration and repetitions

To maximise the learning speed, each target presentation trial was set at a maximum of 3 (simulated) seconds, this being sufficient for even a slow hand speed to reach all allowable target positions. However, if the target was struck the trial was curtailed after 1 further second to minimise the average trial duration over time, an important factor in overall learning duration where numerous trials must be performed with this slow model.

Each target presentation trial is repeated 5 times, looking to exploit the effects of raising reliability under signal dependent noise (see section 4.9.4). As discussed earlier

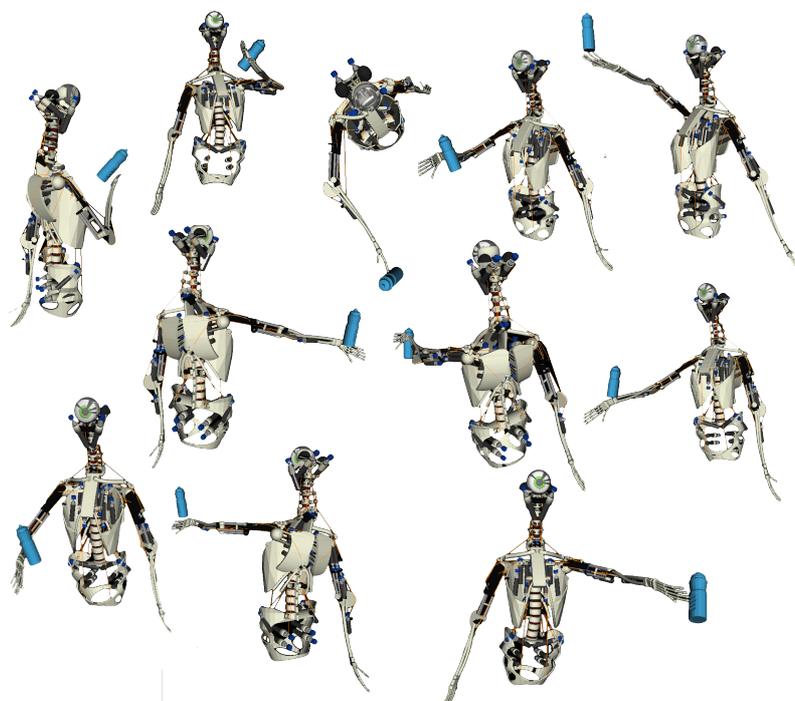
(see section 4.6.2), the noise level parameter  $k$ , was set to  $k=0.2$ , corresponding to the lower values observed in nature (Hamilton et al. 2004).

Four long extended learning trials were undertaken where the main learning cycle (Figure 20) was set to iterate continuously while the reward issued was monitored. On each it was found that learning (as judged by the reward distribution pattern) plateaued in the region of 800 target presentations (see Results), trials were therefore curtailed at 1000 target presentations.

## 5.4 Results

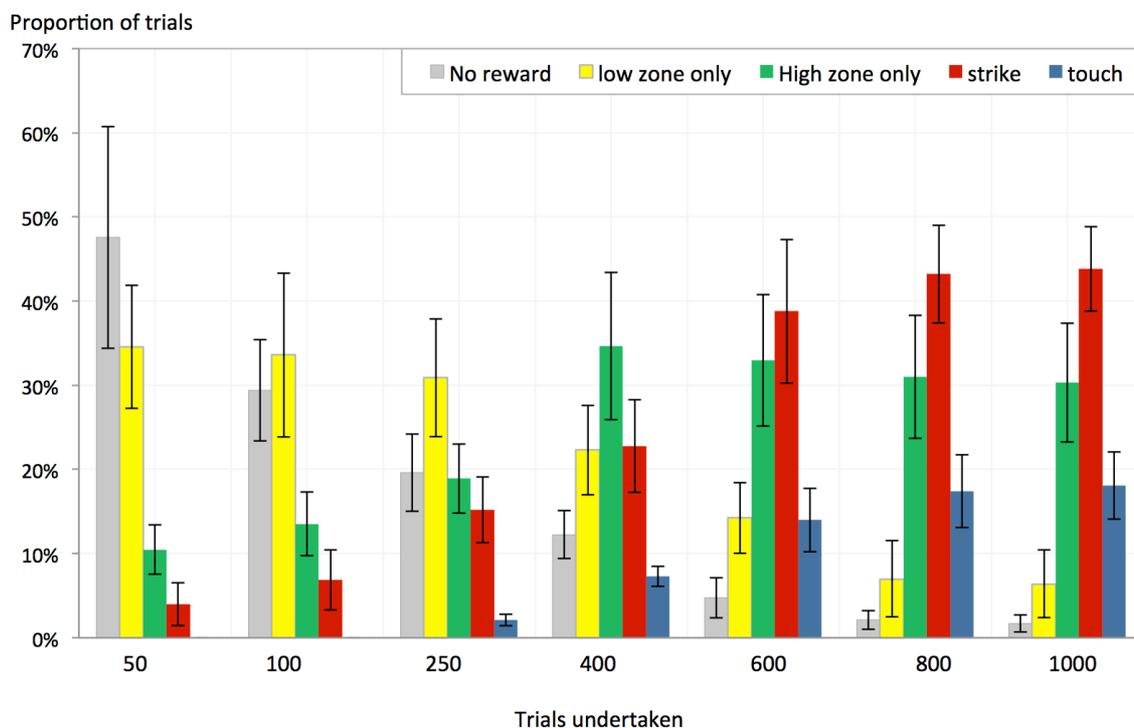
### 5.4.1 Reaching to the target

Results obtained when reaching to a random target location from the same starting state are generally encouraging. To summarise, after sufficient trials the robot was able to, at least, strike the bottle in a majority of target locations, although higher locations were less successfully reached. Figure 23 illustrates a range of examples of successful reaching, however the outcome can be better appreciated by viewing the video of the robot's reaching in action, this can be found at <http://tinyurl.com/ECCE-RL1>.



**Figure 23. Examples of successful reaching to the target**

Screenshots showing the robot model striking the target object under muscle co-activation based motor control acquired via reinforcement learning



**Figure 24. Distribution of trial outcomes at six stages of learning**

The distribution results are shown as a percentages of the trials undertaken during each phase.

Each data point shows the mean value across the 4 extended learning trials. The error bars show the standard deviation.

A trial constitutes testing an action intended to reach to a target presented at a randomly generated location.

The outcome categories are defined as:-

*No Reward* - no reward was awarded during trial

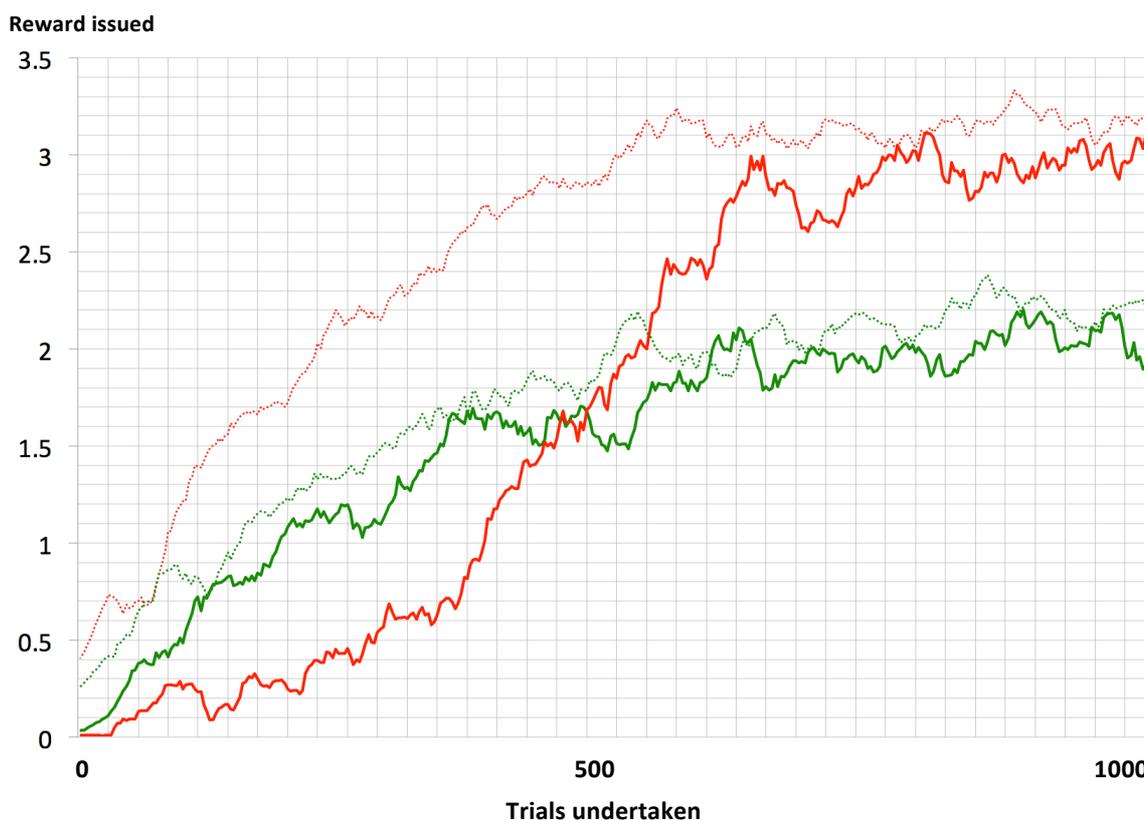
*Low Zonal Reward* - awarded low reward (< mean zonal reward across all trials). No strikes/touches.

*High Zonal Reward* - awarded high reward (> mean zonal reward across all trials). No strikes/touches.

*Strike* - the target was struck by the hand ( hand speed >  $0.1\text{ms}^{-1}$ ) during the trial

*Touch* - the target was "touched" by the hand ( hand speed <  $0.1\text{ms}^{-1}$ ) during the trial, i.e. hand was slowed for potential grasp

Figure 24 shows how, over the lifetime of the four extended learning trials, the outcome of reaching actions changed across 5 categories (no reward, low and high rewards via the proximity zone, striking the target and touching the target). For all the extended trials, the distribution pattern of outcomes settled after around 800 target presentations. After 1000 presentations, mean [*strike*, *touch*] rate - i.e. the hand reaching to either strike or touch the bottle - was [43.2%,18.1%] with standard deviations [5.3%,4.1%] and only failing to obtain any reward at all on 1.7% of attempts on average. However, the ability to just touch the target, i.e. slow the hand speed at point of striking to less than  $0.1\text{ms}^{-1}$  (the point considered sufficiently slow for a reliable grasp to occur) is rather less well developed at a mean of only 18.1% of trials (4.1% s.d).



**Figure 25. Average reward by type issued per trial over 1000 trials**

For each type of reward, the 20 trial moving average is graphed, showing the shift in reward awarded over the course of the learning.

GREEN SOLID: Zonal proximity reward awarded to actions estimated from the presented target location by the policy.

RED SOLID: Strike/Touch reward awarded to actions estimated from the presented target location by the policy.

GREEN & RED DOTTED: Zonal (green) and strike reward (red) awarded to new actions during their re-assessment trials against a better suited problem state (i.e. target location).

Study of the successful touches suggests that they appear restricted to a subset of amenable locations. Analysis of the relevant actions suggests that these match the cases where the hand is able to approach the target with the first co-activation alone, leaving the second to take a greater slowing role over a corrective guiding role.

Figure 24 also shows how the primary reward driver shifts consistently from low scoring zonal only, through high scoring zonal before becoming dominated by target strikes or touches. To show this transition in more detail and to also illustrate the effect of reclassifying new actions with a revised best problem state, we include a detailed plot of reward (amount and type) issued per trial over one of the extended learning trials completed (Figure 25). Mean zonal and strike reward issued to estimated actions are initially zero or close to zero (red and green plots), implying that primarily exploration is occurring. We can see that reward-led learning begins through the initial

strike and zonal rewards awarded to new actions during their re-assessment trials against a better suited problem state (dotted plots). As expected, this reward settles as the policy improves the matching of action to problem state. This is also confirmed by the eventual convergence of each form of reward. The graph also confirms that it is zonal reward (green line) that begins the reinforcement of actions estimated by the policy and provides a reward gradient to the commencement of strike/touch rewards.

#### **5.4.2 Primary issues encountered**

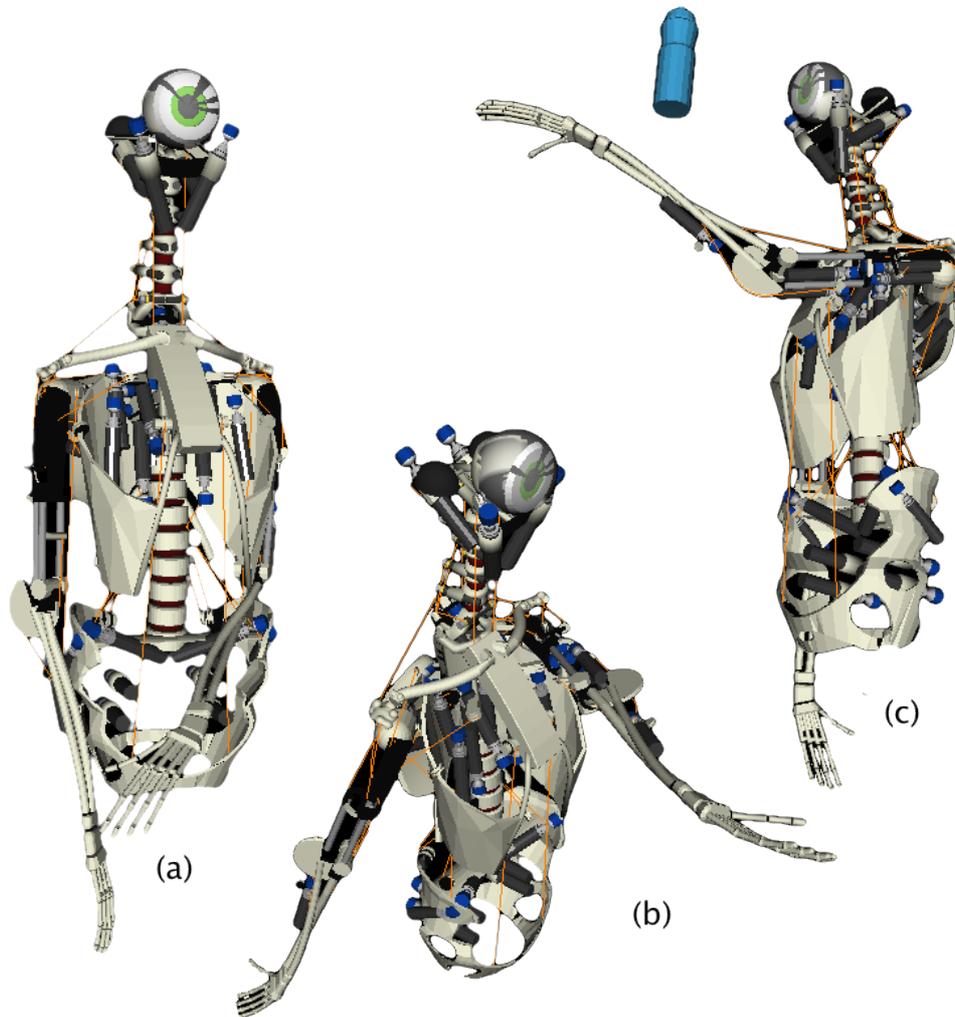
Before embarking on further detailed results analysis we will now acknowledge the problems and issues encountered during the initial reaching experiments; most of these appeared to be model-related.

##### *5.4.2.1 Model issues*

The slow simulation speed (only real time) meant that each of the four extended learning trials of 1000 target presentations lasted approximately 17 hours. Running the model faster than real time would require the GPU acceleration of the physics engine, the machine used for these experiments is already a very powerful high end PC. Since this acceleration remains unavailable at time of writing, and the real robot would always run at real time in any case, we consider means to speed the learning rate by taking more information, more intelligently from trials undertaken (see Future Work section 5.7.4 at the end of this chapter).

Secondly, for some attempted movements the arm and the body can collide and become entangled (Figure 26a). This appears to be a consequence of the default pose in which the robot model was designed.

Finally, for certain target positions, particularly those high and wide, after a reaching attempt the robot is sometimes left in an “unrecoverable” state with the arm twisted upside down and stuck (see Figure 26b for example), i.e. a state from which the muscles are unable to extricate it. This is not an issue for this experiment, but would likely cause issues for a more extensive experiment in which, after reaching, the robot had to go on to attempt another task, such as grasping and moving an object, or simply executing another reaching action.



**Figure 26. Specific model and control issues encountered in learning**

- (a) Arm becomes lodged under ribcage, the muscles are unable to extricate the arm, we refer to such a state as “unrecoverable”.
- (b) Left arm rotates to become twisted upside down
- (c) Fails to reach a high target

#### 5.4.2.2 *Non-model issues*

Even after learning has essentially plateaued, the highest targets were often still missed, although the zonal reward region is entered (see Figure 26c for example). As these movements are beyond the set of initial actions, they are not readily reached by weighted combinations, however the mutation creep added to new actions would be expected to eventually bring these zones fully into range.

#### 5.4.3 **Looking for emerging signatures of optimality**

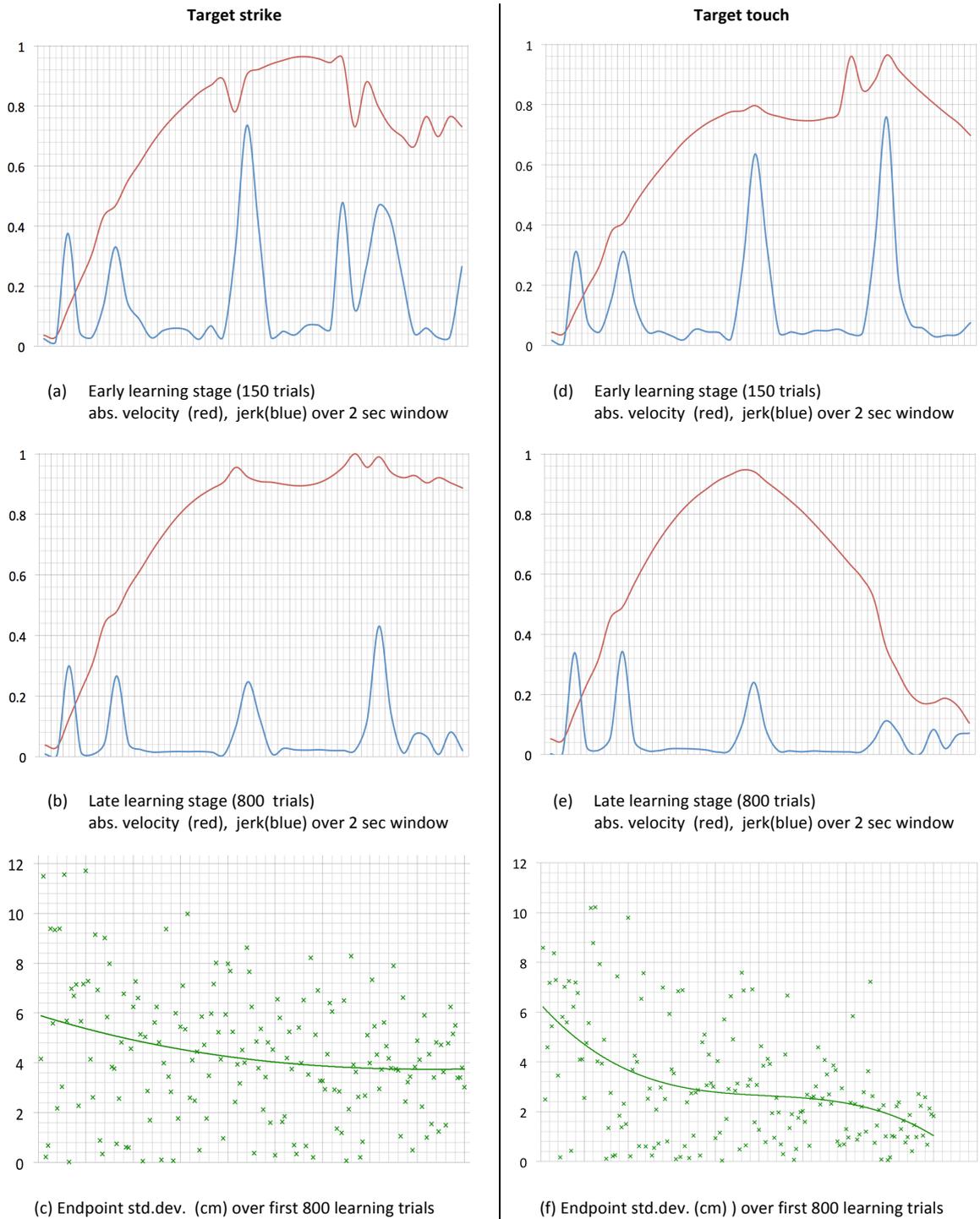
We consider three quantities to judge whether the learning exhibits the optimality behaviour predicted from the use of repeated trials under RL. Firstly, increasing

smoothness of movement by considering the amount of jerk present. Secondly, increasing reliability under signal related noise by considering endpoint variance. Finally we consider whether the velocity profile of the hand moves over the lifetime of the learning towards the stereotype bell curve observed in nature for reaching to a location followed a voluntary halt, for example to perform a grasp. As discussed, this shape is predicted by an application of optimal control using minimisation of endpoint variance as the cost function (Harris & Wolpert 1998; Miyamoto et al. 2004), and also predicted when using minimum jerk as the cost function (Suzuki et al. 1996).

Figure 27a and 23b show a typical example of velocity profiles obtained for the hand speed and jerk (first derivative of acceleration). The figures correspond to the same region at early and late stages of learning (150 trials and 800 trials). To obtain an indication of reliability changes we also plot the variance of the endpoint location obtained during trial repetitions for the same target region over the course of the learning (Figure 27c). We observe, as predicted, a reduction in jerk and a consistent increase in reliability (reducing variance) suggesting there is an underlying optimising process at work with, potentially, a cost function of maximum reliability that was predicted by the use of reward based RL with repeated trialling. However, from the velocity profile, there appears to be little shift toward a bell-curve shape over the course of the learning.

We therefore consider the profiles for one of the target regions where the controller has learnt to slow the hand in the vicinity of the target. The regions were identified by their higher proximity reward and target-touch reward. Here we see (Figure 27(d-f)) that alongside the changes in jerk and reliability we also see the emergence of, subjectively, a more bell-shape velocity profile as the hand is accelerated toward the target then slowed during the second muscle co-activation.

To eliminate subjectivity we compare all trials for targets presented within these "touch" regions to the same number of targets selected randomly outside these regions. Alongside reliability (variance) we also plot conformance to a stereotype bell-curve. To achieve this we apply the chi-squared test (Snedecor & Cochran 1989) which provides a comparison measure between an observed and proposed distribution function. In this case we employ the standard PDF curve (probability density function)



**Figure 27. Signatures of optimality - reaching profiles and reliability change**

The two columns relate to target regions where (left) a strike action was learned and (right) a touching action was learned. Figs. (a)(b)(d)(e) show absolute velocity (red trace) and jerk (blue) over a 2 second window starting when motor voltages are applied. The traces are shown as a fraction of the maximum values reached over the learning cycle for that quantity. For figures (c) and (f) each point (green cross) shows the standard deviation of the hand endpoint in cm for 5 repeated trials of same target under signal dependent Gaussian noise. The points are plotted every 10 target presentations over the first 800 learning trials. After this point, learning has been shown to have settled (see Figure 25). The trendline shows the best fit 3<sup>rd</sup> order polynomial.

as a simple mathematical approximation to the bell-curve distribution observed in human motor experiments.

Figure 28a shows “bell-curve” probability density functions (PDF) approximating the velocity profiles measured from humans, such as the saccade profiles measured by Collewijn et al (1988). Figure 28b shows how the approximating function is fitted to two example velocity profiles (normalised) before applying the chi-squared comparison test. The test is scaled so that profiles such as the blue trace score around 0.6 whilst profiles such as the red score around 0.01.

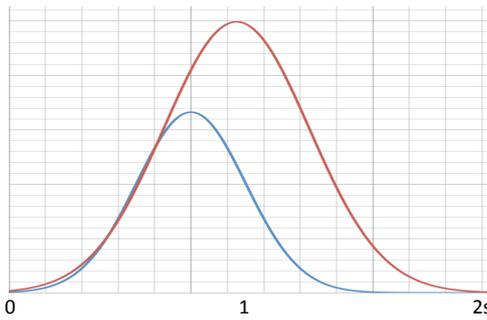
The results of the chi-squared comparison are generally consistent with the two individual trials examined previously in Figure 27. The regions in space where the robot learns to achieve a touching action show an overall trend towards a bell-curve shape (Figure 28c blue), and an increase in reliability (Figure 28e) – reducing endpoint standard deviation from around 8cm down to little more than 2cm. Regions outside these show little move towards a bell-curve (Figure 28c red) and a much smaller gain in reliability (Figure 28f) – reducing endpoint standard deviation from around 8cm down around 5cm.

To test if the results are due to the effect of signal-dependant noise we repeat a full learning trial using a fixed level of Gaussian noise set at half the maximum noise level output by the signal-dependent trials. The results are shown in Figure 28d and exhibit a minimal reduction in endpoint variation.

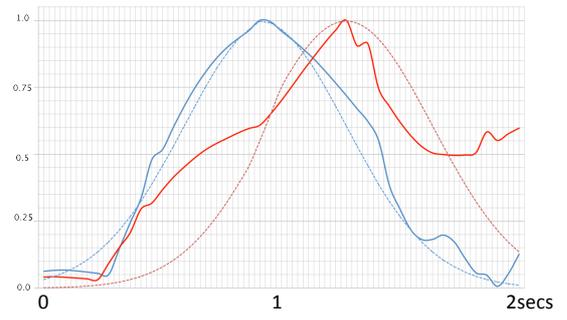
In conclusion, for those target regions where the robot has learned to significantly slow the hand on arrival we do observe that under reinforcement learning a migration towards the stereotype bell-curve “signature of optimality” velocity profile. Across all targets regions we also observe an increasing smoothness of movement (reduction in jerk) and an increase in reliability. As predicted by the optimality theories put forward by Harris and Wolpert (1998), these results applied when adding signal-dependent Gaussian noise, but not for fixed-level Gaussian noise.

#### **5.4.4 Exploiting biomechanical structure**

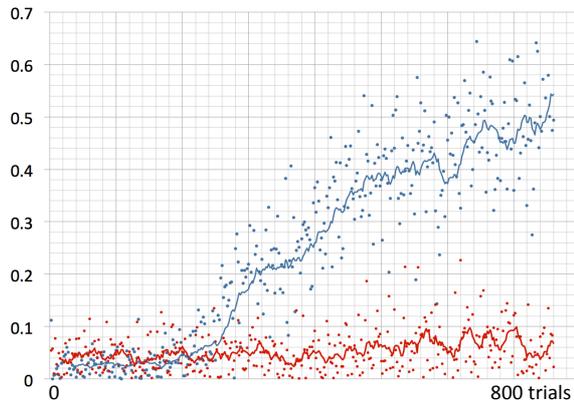
We consider whether and how the actions learned are exploiting biomimetic aspects of the robot structure, not available for a conventional robot with stiff joint-based actuation. We consider three aspects in particular; muscle (motor cable) compliance,



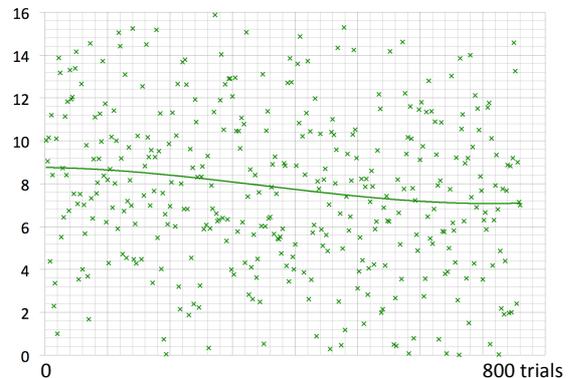
(a) "Bell-curve" probability density functions (PDF) approximating velocity profiles measured from humans (e.g. those plotted by Collewyn et al, 1988)



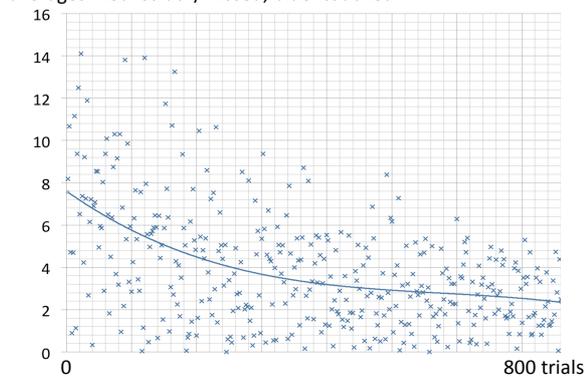
(b) PDFs (dotted) fitted to normalized velocity profiles (solid) for purpose of applying chi-squared comparison test. The test is scaled so that profiles such as the blue trace score around 0.6 whilst profiles such as the red score around 0.01.



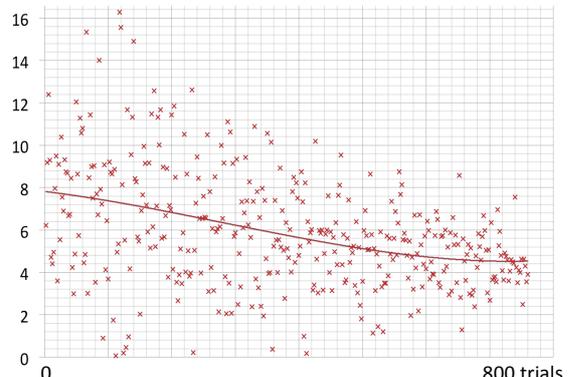
(c) Chi-squared comparison test of PDF bell-curve fitted to hand speed, plotted over first 800 trials. Each point plots the test score for a single trial target. Trace lines shown are the 10 point moving averages. Red=struck/missed, blue=touched.



(d) The effect of using fixed, not signal dependent Gaussian noise. Across first 800 trials we plot the endpoint std.dev(cm) for 5 attempts to reach a target selected.



(e) Across the first 800 trials we plot the endpoint std.dev (cm) for 5 attempts to reach a target selected within those regions where a touch is eventually learned.



(f) As (e) but for targets outside the "touch" regions. Sample size matched to (e) by selecting the first qualifying trial immediately after an entry in (e) was made.

**Figure 28. Optimality under noise; changes in reaching reliability and conformance to bell curve stereotype during learning**  
 Across the first 800 trials we compare all trials targeting "touch" regions to the same number selected randomly outside these regions.  
 Fig.(c) shows the changes in conformance to a stereotype bell-curve obtained by applying chi-squared test (Snedecor & Cochran 1989) using a best-fit PDF function as the "expected" values. To obtain a useable balance between good and poor conformance we scale the chi-squared test so that the more bell-like curves observed emerging at the end the cycle (fig.C, blue trace) score around 0.6 in the test whilst poor conformance such as fig. (c) red trace score around 0.01.  
 The probability density function (PDF) was chosen as a function approximation (fig. a) to the bell-curve stereotype observed in motor action experiments such as Collewyn et al, (1988).  
 Fig. (d) The effect on endpoint variance of using fixed, not signal dependent Gaussian noise.  
 Figs. (e) and (f) show how reliability increases (endpoint variance decreases) across a learning cycle, comparing regions where a touch action is eventually learned (e) to other regions (f). Each point plots the standard deviation in cm of 5 repeated attempts to the same trial target location.

the use of full-body dynamics and evidence of so-called “morphological computation” afforded by specific biomechanical structures such as the floating shoulder blade.

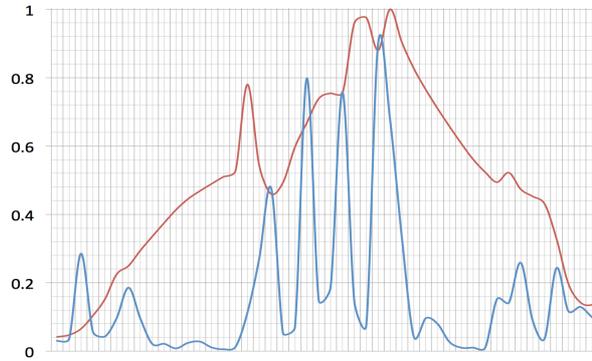
#### 5.4.4.1 Compliance

The main source of compliance in the robot are the sections of elastic shockcord employed in the muscle cable attachment. However, as spring forces play in part in all movements it is difficult to quantify the degree to which is being specifically exploited. We therefore examine its effect on reaching control by testing a null hypothesis, namely that applying the learning to a model with little or no compliance produces the same behaviour, including equivalent learning rates, reward levels and reliability scores.

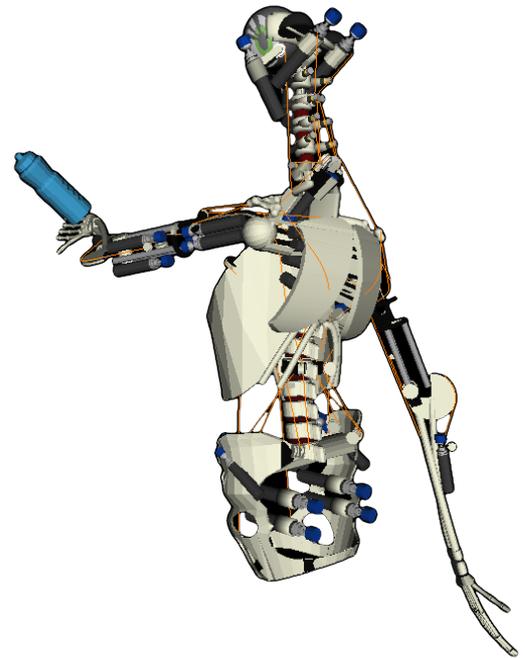
We employ a modified model where the spring constant of the elastic shockcord is raised by approx. 100 times from that of the physical material. Note that, beyond this point, we find that the imbalance of forces triggers the physics simulation to “explode”. This is a known issue with impulse based simulation when managing the scaling of very disparate forces.

After an equivalent number of learning trials, We find that reaching for targets has been learned but with both a markedly reduced success rate (Figure 29c) and reduced reliability (see endpoint variance graph Figure 29d). We also see that the transition within a movement between the two muscle co-activation stages exhibits markedly higher jerk elements (Figure 29a) than the comparable traces obtained for the compliant model (Figure 27e). This jerk can also be discerned visually as a distinct juddering (see video available at <http://tinyurl.com/ECCE-RL2> ).

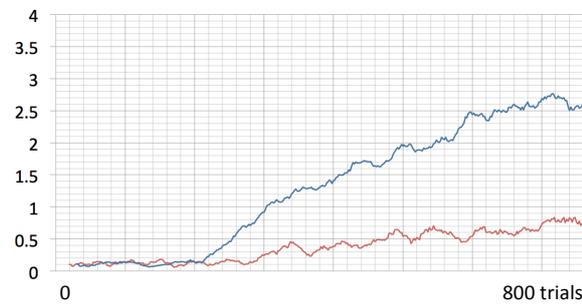
However, although minimised jerk (Suzuki et al. 1996) and maximised reliability (Harris & Wolpert 1998) are indirectly associated as optimising cost functions that predict biological velocity profiles there is no specific causal link suggested, i.e. higher jerk reduces reliability. Rather, improved reliability is associated with reduced motor signal amplitudes which then output correspondingly less noise. This then is a puzzle, until we consider that compliance in muscles affects the power requirements for a change in actuation through its ability to store energy (Lichtwark & Barclay 2010). Measuring the motor signal magnitudes learned for a comparable target location presented to both compliant and non-compliant robots shows that the compliant robot was able to use 23.7% less force in the second stage co-activation. This will therefore



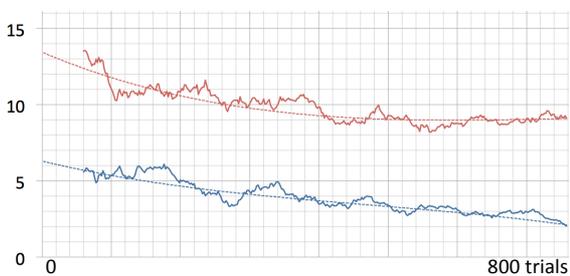
(a) Velocity( red) and jerk (blue) profiles for reaching movements learned in a model with very low (stiff) compliance muscles. A 2 second window is displayed.



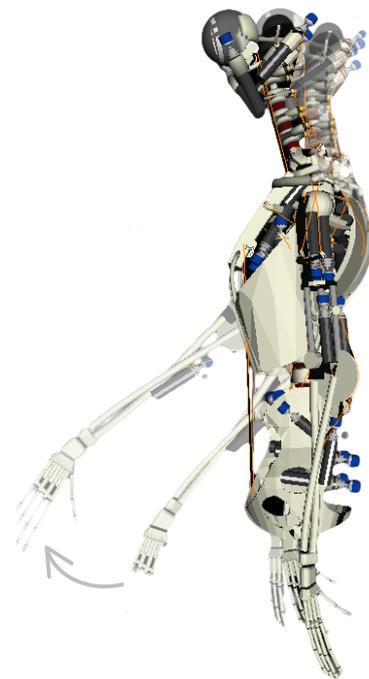
(b) By use of a counterbalancing arm the emergent control leverages dynamics of the full body model



(c) Rolling average of reward per trial over first 800 trials (compliant:blue, non-compliant:red )



(d) Rolling average and trendline of std.dev. of endpoint over 5 attempts per target. First 800 trials. (compliant:blue, non-compliant:red )



(e) Contracting the supraspinatus muscle causes the arm to not only raise outwards but also swing forward. In this way the emergent control leverages morphological computation (via the biomechanical structure) to commence a reaching movement.

**Figure 29. Exploiting the biomimetic aspects of the of structure; compliance, full-body dynamics and biomechanical structure**

produce correspondingly less signal related noise which predicts greater endpoint reliability. In addition, a larger change in force at the co-activation switchover also directly relates to higher jerk (derivative of acceleration) through elementary laws of mechanics ( $F=ma$ ).

We conclude that, while detailed analysis of forces and energy has not been undertaken, the initial results suggest that the control emergent under RL was able to exploit motor cable (“muscle”) compliance as a spring energy store to reduce the force required when switching between stages of a movement, resulting in more reliable outcomes through lower noise, and smoother (less jerky) movement.

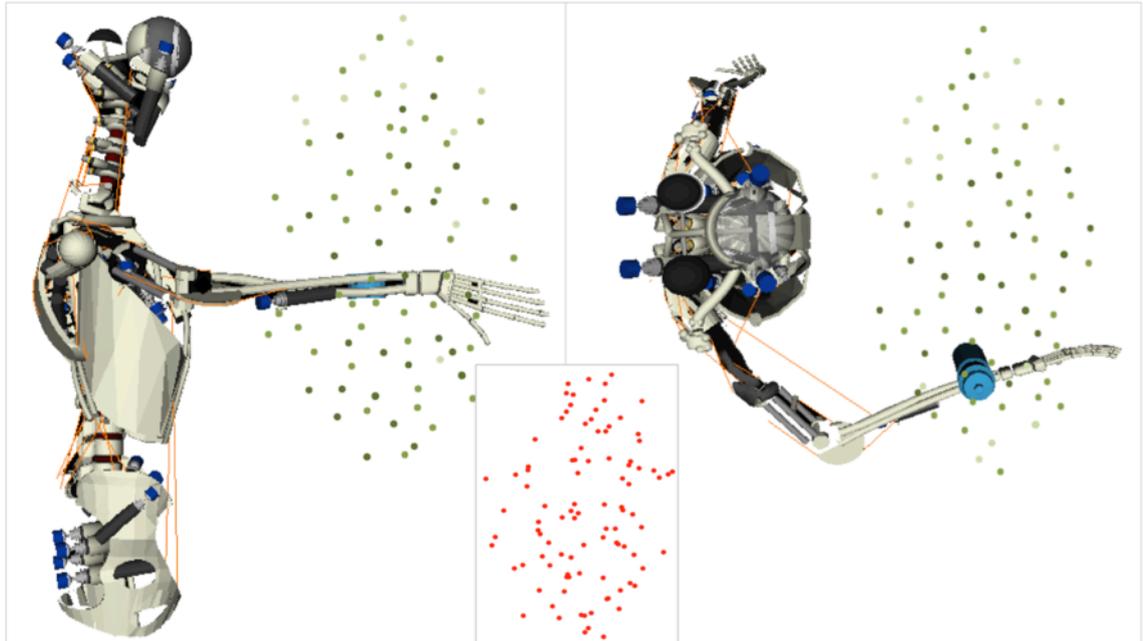
#### ***5.4.4.2 Evidence of exploiting natural dynamics***

Alongside compliance, the emergent control can be seen exploiting aspects of full body dynamics to aid performance, most prominently in the emergence of a reciprocal backwards movement of the opposite non-reaching arm (see Figure 29b) caused by an distinct activation of muscles (non-passive). Triggering the same action while the opposite arm muscles are artificially disabled results in the robot over-balancing forwards for higher target locations. This strongly suggests that this movement was acquired to aid stabilisation through counterbalancing.

A third area of interest is to consider where the emergent control has leveraged natural mechanical dynamic characteristics of the structure to its advantage. Although not simple to quantify, we see a clear example in the use of the supraspinatus muscle that runs across the top of the shoulder. This muscle would be expected to lift the arm in a simple outward direction, yet the emergent control targets this muscle heavily at the start of a reaching action. Testing of this muscle in isolation shows that the mechanical elements of the shoulder, including the free shoulder blade that forms half of the shoulder joint, interact in response to the arm-raising, causing it to swing forward as well as outwards (see Figure 29d). This could be interpreted as an example of morphological computation, where a simple control signal can generate a richer and more useful movement due to the particular dynamic coupling implemented between the muscular and mechanical systems.

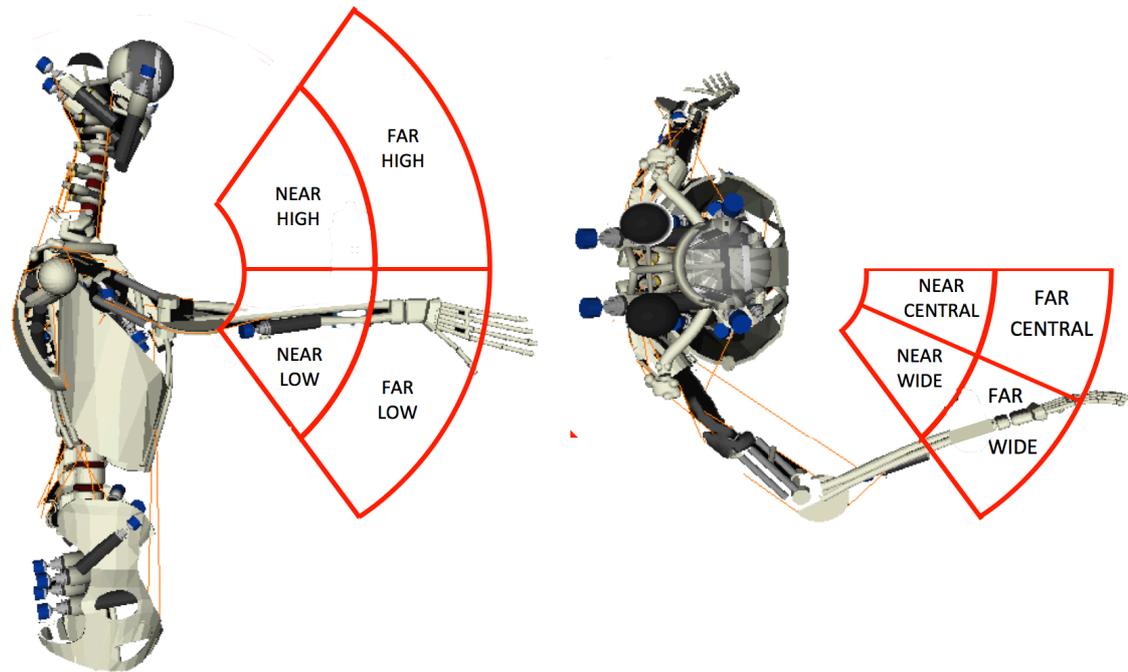
### 5.4.5 Emergent muscle activation patterns and synergies

We now begin to analyse the muscle co-activation patterns and driving signals that emerge over the learning and discuss pertinent aspects.



**Figure 30. Distribution of target locations linked to stored motor plans**

Each dot indicates the target location attached to a stored motor plan at the end of a learning cycle. The reward (high, medium, low) attached to the plan is shown by the boldness of the dot. For comparison, the inset illustrates a random distribution.



**Figure 31. Eight reaching zones defined for muscle co-activation pattern analysis**

The eight zones are labelled as:

LOW-CENTRAL-NEAR, LOW-CENTRAL-FAR, LOW-WIDE-NEAR, LOW-WIDE-FAR,  
HIGH-CENTRAL-NEAR, HIGH-CENTRAL-FAR, HIGH-WIDE-NEAR, HIGH-WIDE-FAR

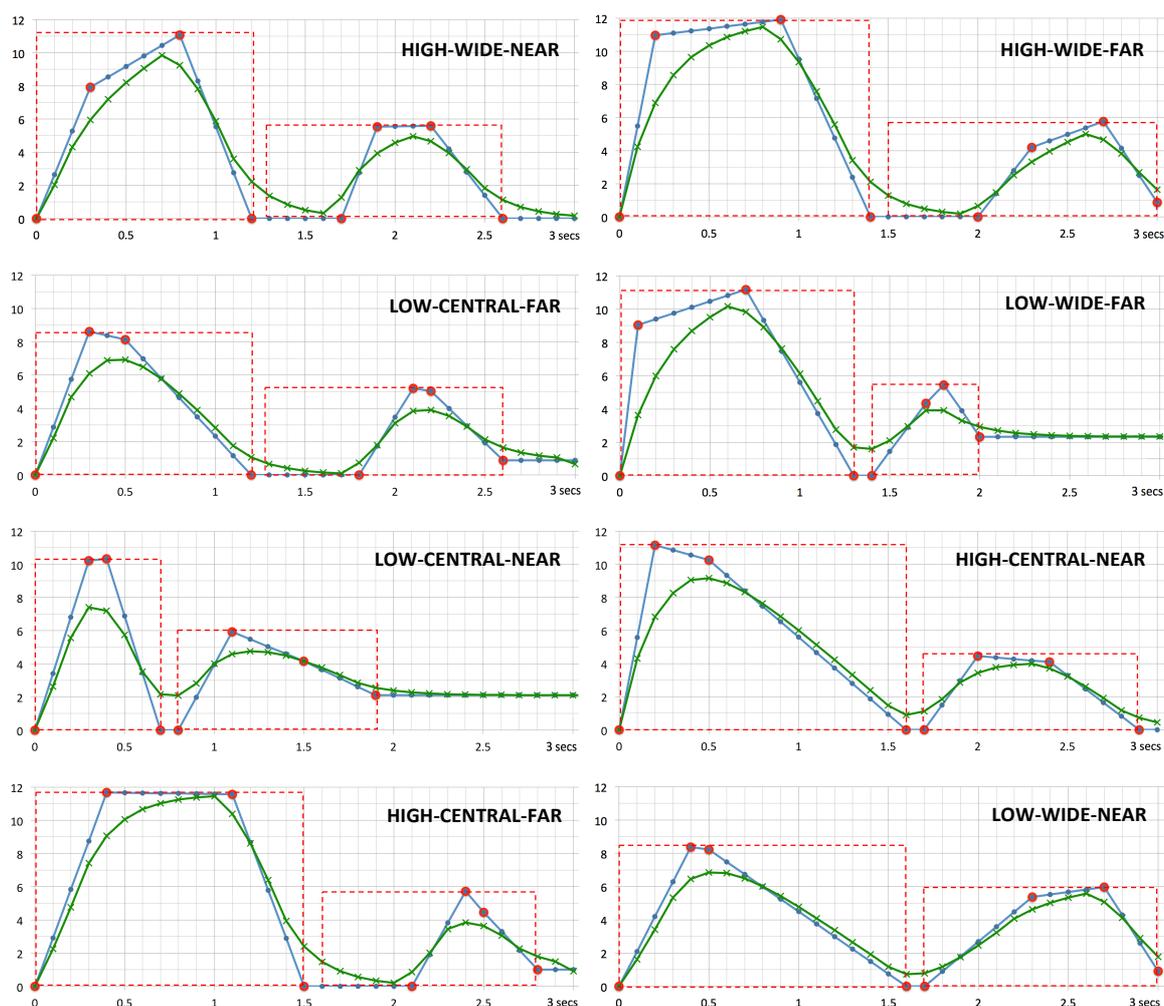
#### 5.4.5.1 *Motor plan distributions by target and reward*

Figure 30 illustrates a distribution of target locations linked to stored SARs after a typical 1000 trial learning period. By considering the average Q values across different Regions shows that more valued plans favour lower and central target positions – reflecting most likely the greater ease of reaching these locations. However, the figure also suggests that the distribution of targets is more evenly spread than a random selection of points (for an example, see Figure 30 inset figure, red dots). This is confirmed by the standard deviation of nearest-neighbour distances which is just 9.3% of the s.d. of the random distribution. This strongly suggests a “winner-take-all” competitive process in action where plans with over-closely located targets tend to block each other’s reward, creating the effect where much of the final set appear to “have claim” over their own small target “territory”.

#### 5.4.5.2 *Emergent co-activation patterns and driving signals*

To study the nature of the emergent co-activation patterns and driving signals, we select for analysis a set of influential (valuable) actions covering a substantial proportion of the reaching zone. We identify the most rewarded action that has an attached target location that is closest to the centre of each cell in a grid of eight zones. These zones are set to reflect all permutations of central/wide, near/far and high/low (see Figure 31 for a plan). The set of 2 muscle co-activations corresponding to each compound action are detailed as weighting graphs in Figure 33, whilst their corresponding driving signal waveforms are shown in Figure 32. We observe that all driving signals sent to the motors (green traces) have converged to a core template shape comprising two smooth activation peaks, although each is shifted relatively in time and amplitude to reflect the distance or effort to reach the target zone. It should be recalled that the parameterisation of the driving signal (see 4.4.1 and Figure 19) does not dictate this outcome as it could describe a far wider range of activation shapes, for example, beginning or ending at a non-zero level.

In fact, there is, interestingly, a distinct similarity between this core double peak shape with the equivalent signals observed in human reaching by Cheung et al (see Figure 5). However the most interesting observation is that the similarity of these driving signals

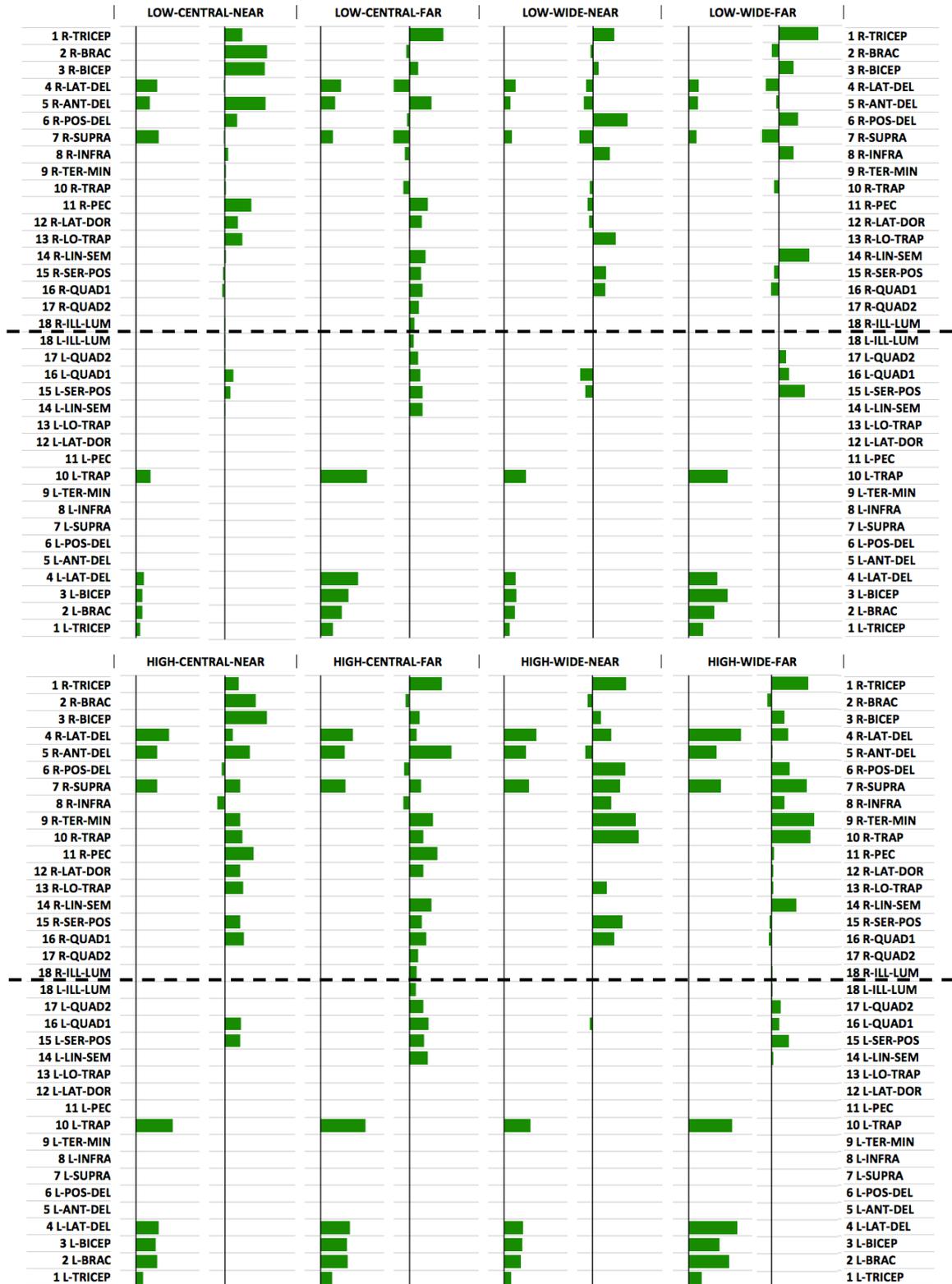


**Figure 32. Parameterised driving signals of most valued reaching actions with targets spread across eight sub-zones**

Pairs of red boxes indicate consecutive parameterised signals each driving a muscle co-activation pattern (see Figure 33) learned as a compound reaching action. The red dots indicate the 4 parameter points shaping each driving signal. The blue traces shows the resultant output muscle signal sent at 100ms intervals. The green traces shows the signal actually sent to the motors after smoothing has been applied through the use of a low-pass-filter.

across zones strongly suggests that it is rather the learned co-activation weighting patterns that are the primary casual element in differentiating behaviour.

Moving on therefore to consider these weighting patterns (Figure 33), the first stage co-activations clearly exhibit the use of a similar set of muscles across all zones, albeit in varying relative amounts. The two main components appear to correspond to behavioural features noted earlier (see 5.4.4.2): firstly the use of the supraspinatus and other shoulder muscles to raise the arm (which we have already noted to cause a forward swing by dint of the mechanical structure) and secondly, the pulling back of the opposite arm discussed earlier that appears to provide some balancing compensation. Second stage co-activations present markedly less commonality in their



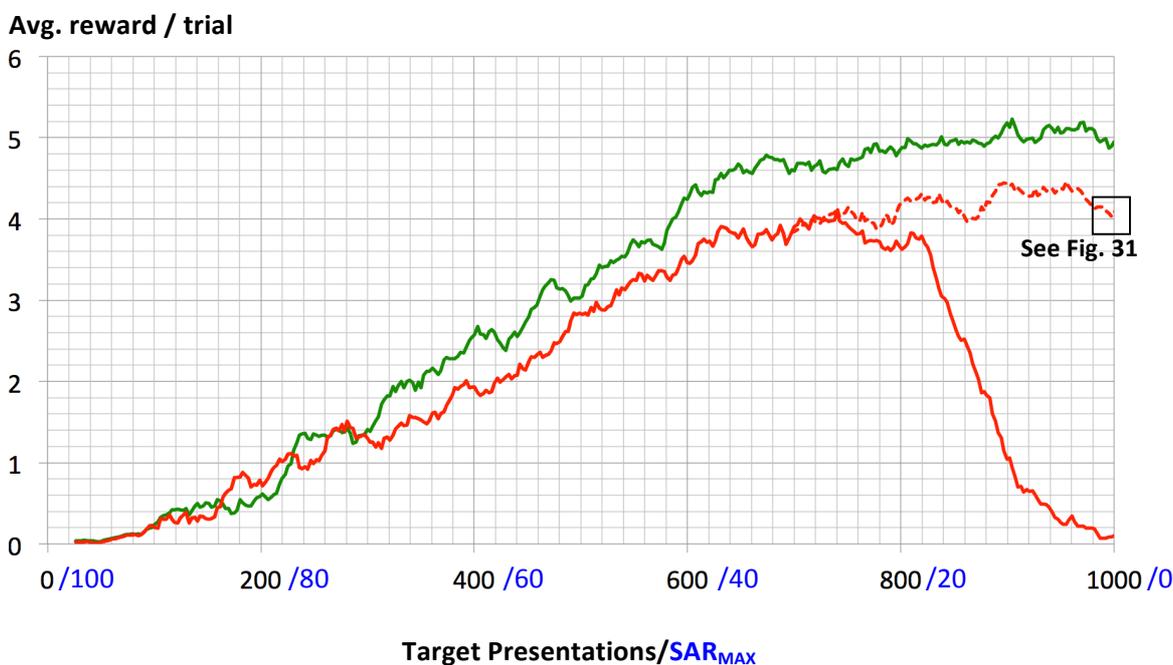
**Figure 33. Muscle-coactivation patterns of most valued reaching actions for a target in each of eight sub-zones**

The figure shows the most valued compound action for each of 8 zones (see Figure 31) as a pair of muscle co-activation patterns that are invoked consecutively for the reaching behaviour. Length of each bar reflects strength of a muscle activation within a particular pattern. The top half (above dashed line) of the pattern relates to the muscles of the reaching side, designated right (R-) for convenience. The bottom half relate to non-reaching side (L-). MUSCLES: 1. Triceps, 2. Brachialis, 3. Biceps, 4. Lateral Deltoid, 5. Anterior Deltoid, 6. Posterior Deltoid, 7. Supraspinatus, 8. Infraspinatus, 9. Teres Minor, 10. Trapezius, 11. Pectoralis, 12. Latissimus Dorsi, 13. Lower Trapezius, 14. Linea Semilunaris, 15. Serratus Posterior, 16. Quadratus (i), 17. Quadratus (ii), 18. Ilio-Costa Lumborum

approach to bringing the hand to the target zone. However, there are some interesting glimpses of common underlying elements; the antagonistic co-activation of bicep/tricep/brachialis to pull the elbow in for central+near targets; the near symmetrical use of back muscles to reach central, far targets; the high co-activation of Teres Minor and Trapezius to reach high, wide targets. In order to identify key common patterns (potential synergies) more clearly we trial two contrasting approaches; firstly, reducing the number of retained actions to a minimum, and secondly, a more formal factor analysis looking to identify *candidate synergy* patterns.

### 5.4.5.3 Minimising number of retained actions

We begin an extended learning period from scratch, comprising again 1000 target presentations. making the significant new alteration that the maximum allowable number of retained actions (SARs) is reduced continually over the course of the cycle starting from the maximum,  $SAR_{MAX}$  set to 100. In the results presented below  $SAR_{MAX}$  was reduced by one for every 10 target presentations.



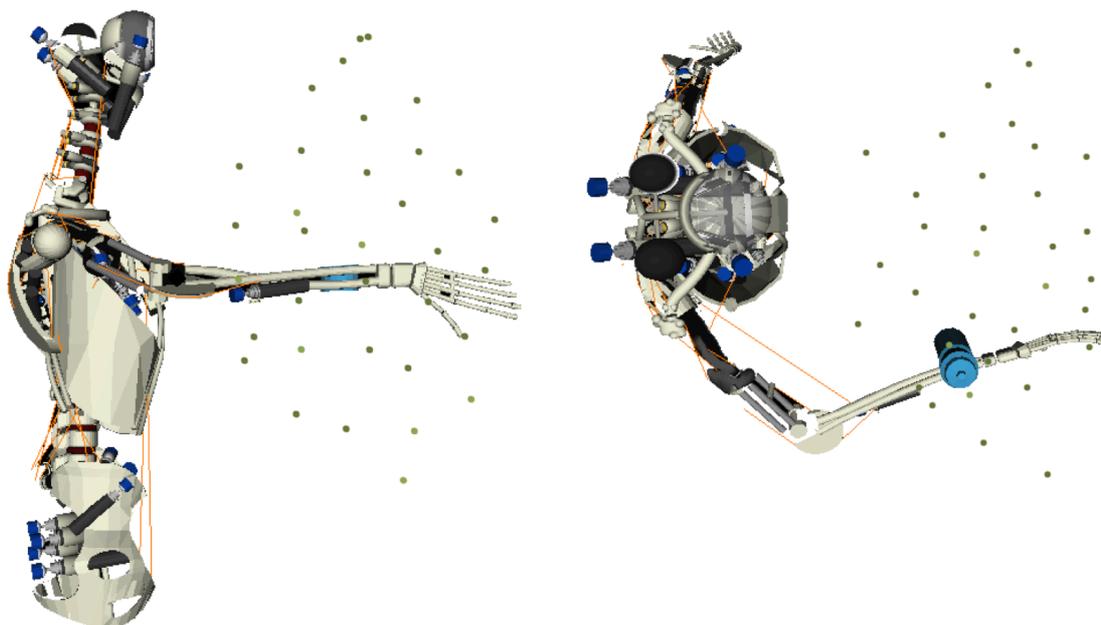
**Figure 34. Effect on learned reaching performance of continually reducing set of stored actions**

Shows the performance of a learning trial where the maximum number of actions ( $SAR_{MAX}$ ) retained by the policy is reduced by one every 10 trials starting from  $SAR_{MAX} = 100$ . Performance is quantified as before by the rolling average (20 trial window) of reward awarded per trial. Note that reward score graphed comprises the total of zonal + strike/touch reward.

GREEN TRACE: Reproduces for comparison reward data obtained previously with  $SAR_{MAX}$  fixed at 100.

SOLID RED:  $SAR_{MAX}$  reduced throughout by one every 10 trials starting from  $SAR_{MAX} = 100$ .

DASHED RED: As solid red trace but reduction is halted at  $SAR_{MAX} = 40$



**Figure 35. Distribution of target locations with a minimised set of stored actions (40 entries)**

Each dot indicates the target location attached to a stored action at the end of an extended learning period where the number of stored actions was reduced every 10 trials from 100 to a minimum of 40 (at 600 trials) before continuing to 1000 trials. The reward (high, medium, low) attached to the plan is shown by the boldness of the dot.

This more aggressive pruning is intended to encourage the elimination of redundancy, and favour instead the maximum reuse of stored co-activation patterns in the weighted combinations generated when a new target is presented. In this way we look to encourage the emergence of distinct and diverse muscle synergies that nevertheless combine effectively in simple linear weightings to produce successful reaching behaviour. We find that (Figure 34), until  $SAR_{MAX}$  is reduced to around 40 actions, reaching performance (measured by the average reward awarded to new actions) remains within 10% of that attained through previous learning trials where  $SAR_{MAX}$  was left at a constant 100. However, below approximately  $SAR_{MAX} = 30$  entries, performance tails off rapidly. Study of a snapshot of muscle co-activation patterns within the actions (when  $SAR_{MAX} = 40$ ) reveals no evidence of the sought-after separation into a set of diverse, distinct synergies. The corresponding “cloud” of target locations (Figure 35) associated with the actions remains even but simply further widespread than previously (Figure 30), albeit focused more to the centre than the extremities. Reward is more evenly spread, largely at a consistent high level. Testing, in isolation, the individual reaching performance of a random selection of 5 stored actions finds that they reach their associated target location more successfully than would be expected of activation patterns representing core common component

synergies, which we suggest should, by definition, function effectively primarily in weighted combination and not in isolation. Figure 35 also suggests a reason for the failure of stored actions to diversify as hoped. With our learning algorithm, plans are heavily associated with a single target location and close matches with a new problem target will be heavily rewarded if successful. There is thus little chance of actions emerging that, when acting as the dominant weighted element, do not perform well in reaching their “contract” target. We nevertheless note in passing the useful result that the performance can be retained (within 10%) with significantly less stored actions, namely 40 entries instead of 100.

As useful synergies have not emerged from this learning-based approach, we therefore move on instead to a second, more direct but less elegant, approach that seeks to identify *candidate* synergies using analysis techniques, aping the successful process employed for task in biological motor experiments (e.g. Cheung et al. 2009, see Background, section 2.5.5).

#### 5.4.5.4 *Analysis of muscle signals for candidate synergy extraction*

We analyse the muscle co-activation patterns learned (Figure 33), searching for common pattern fragments that may, in weighted combination, underpin them. Note that, although these distinct recurring fragments are commonly referred to as “muscle synergies” by many of the biological studies that have located such common elements through analysis, these are, however, often primarily empirical studies (e.g. Cheung et al. 2009; D’Avella et al. 2003; Hart & Giszter 2004; Ivanenko et al. 2005; Ting & Macpherson 2005; D’Avella & Tresch 2002). Thus they do not necessarily show unequivocally (e.g. anatomically) that these patterns are being explicitly employed blockwise as “true” synergies by motor centres - only that complex data can be well explained by this simple assumption. We will therefore refer to our own analysis findings as *candidate synergies*. Further investigation will be required to show if they can function as true synergies.

For a suitable technique to locate potential candidate synergies we refer to the comparative study “*Evaluation of matrix factorization algorithms for the identification of muscle synergies*” (Tresch et al. 2006) where six analysis techniques (including PCA and ICA) were tested on their ability to accurately extract known synergies from test data comprising weighted combinations. From the recommendations of this study we select

factor analysis (FA) for our trial as the study found it to yield high accuracy whilst being readily available(via the *factoran* Matlab function). This was applied to 200 18-muscle co-activation patterns (100 SARs  $\times$  2 co-activations each) that were learned and stored by the policy after 1000 target presentations.

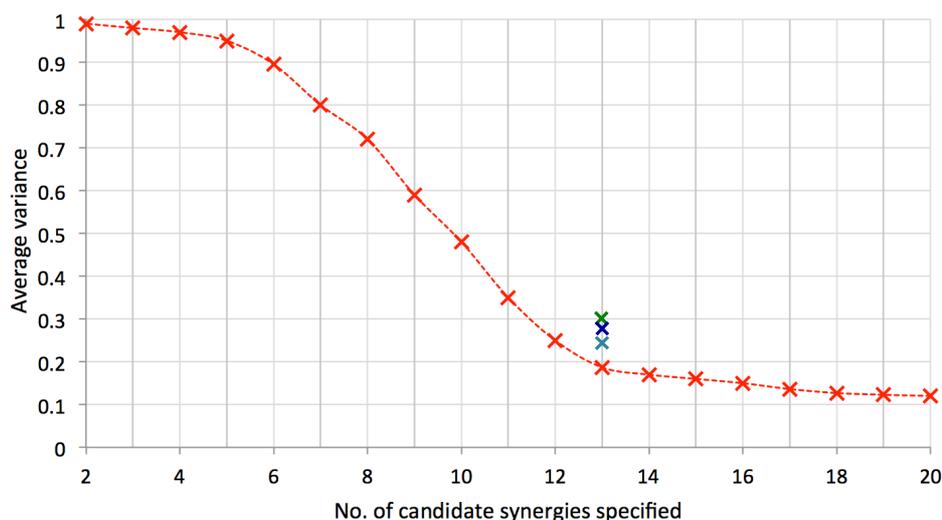
As the number of underlying candidate synergies is unknown we invoke the *factoran* function repeatedly, whilst varying the *common\_factors* parameter, specifying the number of potential synergies that it should model for. We then plot the specific variance returned (averaged across the data set), this indicates how much of the data is accounted for by a weighted combination of the common factors proposed by the function and how much is considered by the function to be unstructured noise. A variance of 0 indicates that the full data is accounted for, whilst a value of 1 indicates that none is. The results are shown in Figure 36.

The analysis shows that, beyond a minimum of 5 candidate synergies, the function is able to account for a rapidly increasing proportion of the co-activation data as the number of common factors is specified. However, beyond 13 candidate synergies the improvement in data accounted for becomes incremental. We therefore conclude that combinations of 13 underlying candidate synergies can effectively account for a high proportion of the observed muscle co-activation data. The 13 patterns returned by the analysis are shown in Figure 37. In addition, Figure 38 reproduces a selection of the zonal co-activations detailed earlier (Figure 33) and illustrates how they can in fact be closely reconstructed from the 13 candidate synergies using the weightings (“loadings”) returned by the FA analysis.

#### 5.4.5.5 *Candidate synergies or true synergies?*

Whilst we have shown that the 36-dimension muscle co-activations retained with the emergent set of stored actions can be well accounted for by a weighted combination of only 13 candidate synergy patterns, we do not yet claim to have located “true” muscle synergies. A much stronger claim could be made if two tests can be applied.

Firstly, how much co-activation data from alternative learning runs can be accounted for using these synergies? This would indicate how strongly the synergies are intrinsic to the musculoskeletal structure rather than artefacts of a specific and unique ordering of target presentations.



**Figure 36. Co-activation data accounted for by weighted combinations of candidate synergies uncovered using factor analysis**  
 Maximum likelihood factor analysis (FA) was performed on 200 data points comprising (noise-free) co-activation weightings learned for 36 muscles. The averaged variance returned from the *factoran* factor analysis function (y-axis) indicates how much of the co-activation data can be accounted for by a weighted combination of  $N$  (x-axis) candidate synergy patterns proposed by the function and how much is considered by the function to be unstructured noise. A variance of 0 indicates that the full data is accounted for, whilst a value of 1 indicates that none is. The 3 additional points  $\times\times\times$  plotted indicate the variance scores of the same 13 candidate synergies matched against the co-activation data from three alternative learning trials.

**NOTES:**

Following recommendations from Tresch et al (2006), analysis was carried out using the *factoran* function in Matlab, specifying *varimax* rotation and *wls* (weighted least squares) method to estimate the common factor loadings (weightings). The number of factors that can be assessed with FA is limited by the dimensionality of the data in this case 36 dimensions sets a limit of 28 common factors.

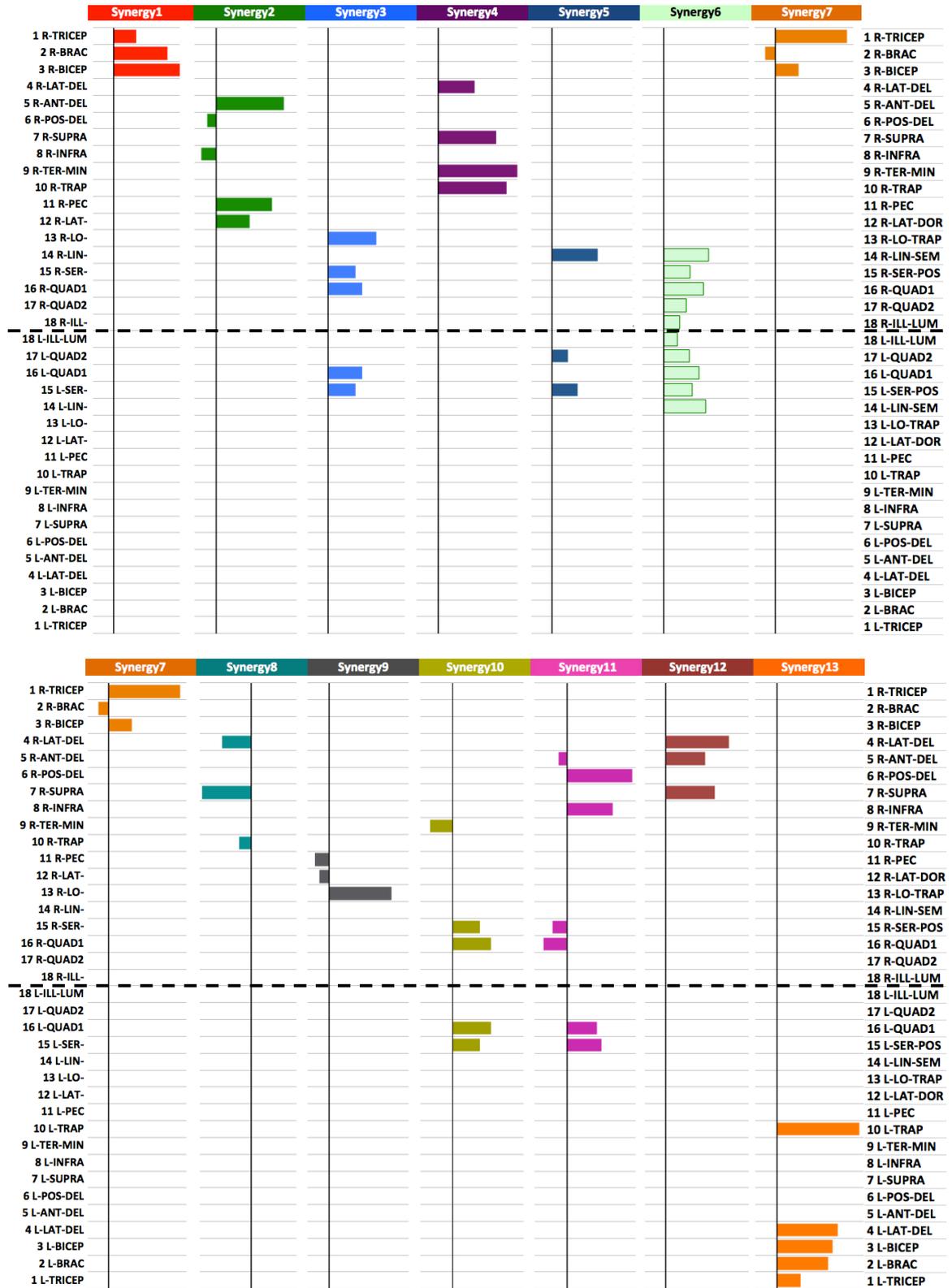
The function is instructed to return a set of  $N$  estimated *common factors* (candidate synergies), and for each data point, a vector of  $N$  *loadings* (weightings) that best reconstruct the data point from the common factors and a *specific variance* measure (indicating the fit of the proposed model to the data point).

Note also that to maximise analysis performance the co-activation data was provided to the function noise-free, although, as discussed, artificial signal dependent noise was added in the robot control trials. Tresch et al (2006) also test FA analysis performance for real data where noise is unavoidable, however, this did not apply in our case.

Secondly, as discussed earlier, if we can show that these candidate synergies can be effectively employed directly as units in what would be, in effect, a low dimensional ( $n=13$ ) controller, then we can claim to have demonstrated a method to construct a synergy-based reaching controller for a musculoskeletal biomimetic robot. We therefore consider each of these tests in turn.

#### 5.4.5.6 Commonality of candidate synergies across extended learning trials

To what extent do the same set of candidate synergies emerge from any learning trial? Close similarities would suggest that the synergies are linked to the particular dynamic



**Figure 37. Thirteen muscle synergies extracted by factor analysis of muscle co-activation patterns**

Weighted combinations of only the 13 candidate synergies illustrated can account for a high proportion of muscle co-activation patterns generated by the controller following a 1000 target presentation learning trial.

MUSCLE KEY: 1. Triceps, 2. Brachialis, 3. Biceps, 4. Lateral Deltoid, 5. Anterior Deltoid, 6. Posterior Deltoid, 7. Supraspinatus, 8. Infraspinatus, 9. Teres Minor, 10. Trapezius, 11. Pectoralis, 12. Latissimus Dorsi, 13. Lower Trapezius, 14. Linea Semilunaris, 15. Serratus Posterior, 16. Quadratus (i), 17. Quadratus (ii), 18. Ilio-Costa Lumborum

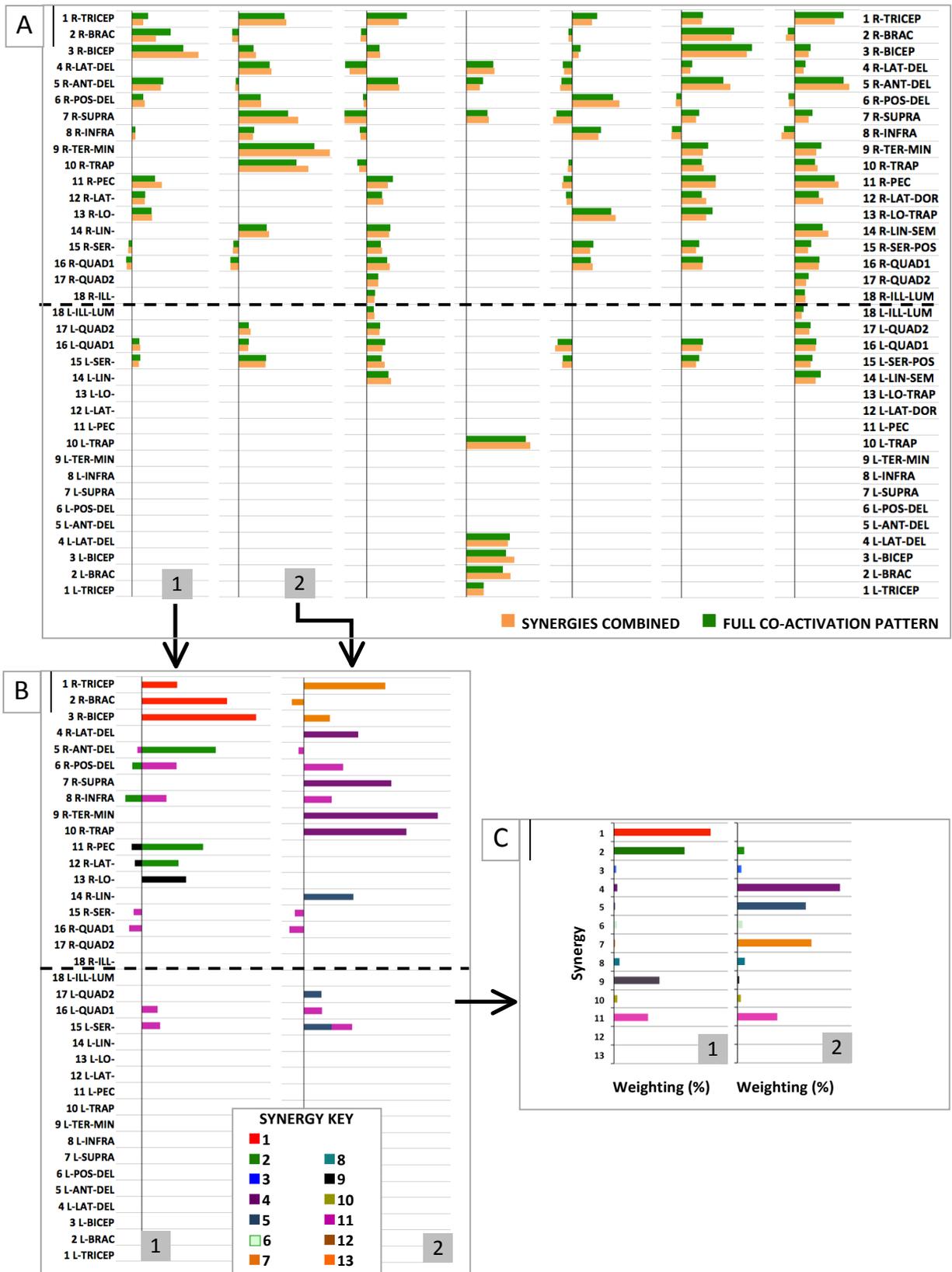


Figure 38. Reconstruction of co-activation patterns from 13 extracted candidate synergies

(A) Selection of co-activation patterns reproduced (from Figure 33) alongside their reconstruction from 13 extracted synergies.

(B) Detail of 2 co-activations' reconstruction from weighted synergies. For clarity, minimally weighted components are neglected.

(C) Breakdown of same 2 co-activations into set of weighted synergies comprising the best-fit reconstruction.

structure under control, rather than comprising an artefact of the learning approach or being biased heavily by the random direction taken by early steps into the state space.

Taking the co-activation data emergent from the three further extended learning trials available, we apply the same weighted least squares (*wls*) method employed in the factor analysis. This provides the variance data that indicates the fit of the co-activation data weighted combinations of the supplied set of 13 candidate synergies. Where the original data scores an average variance of 0.191, the data from the alternative trials score 0.245, 0.282 and 0.298 respectively (see additional points plotted on Figure 36).

The closeness of the fit for these synergies implies that these controlling synergies may indeed be considered expressions of the dynamics of the musculoskeletal structure - within the constraints of a reaching task - and supports the principle of the findings of the study of Berniker et al (2009) which directly analysed a controlled structure's dynamics to locate effective synergies for use in a low dimensional controller - using a "balanced truncation" approach to ascertain dimensions with the most influence. Their results suggest that if our 13 candidate synergies do express structural dynamics of the robot model then a low (13) dimensional reaching controller appears a distinct possibility and would confirm the utility of these candidate synergies as, at least, one possible set of "true" fixed synergies available as units for potentially simplified acquisition of effective reaching control.

## **5.5 A reaching controller based on fixed synergy units**

### **5.5.1 Introduction**

We test a low-dimensional controller based on the 13 fixed-weight candidate synergy units identified using factor analysis of co-activation data (see section 5.4.5.4) and compare its learning rate and reaching performance to our original controller that was based on locating effective co-activations of individual muscles. Furthermore, in order to test to what degree the candidate synergies are generic or task specific we apply the controller to a number of task variations.

### **5.5.2 Method**

This low-dimensional controller could potentially take a range of forms - for example, the simplification of using whole synergies as units may bring the acquisition of

control within the range of standard generic reinforcement learning methods such as TD learning (see methods review in the Background chapter). However, for now at least, we are specifically interested in direct comparison with our original controller, we therefore retain the design of controller and learning algorithm (detailed in the original method), but simply change the units of activation to be fixed muscle synergies rather than individual muscles. New co-activation patterns thus become weighted combinations of these 13 synergies rather than combinations of co-activation patterns of the 36 separate muscles – thus simplifying the search by 23 dimensions. All other elements of the original methods are retained. The controller is trialled with the same reaching task as the original. As before, learning is continued for 1000 random target presentations.

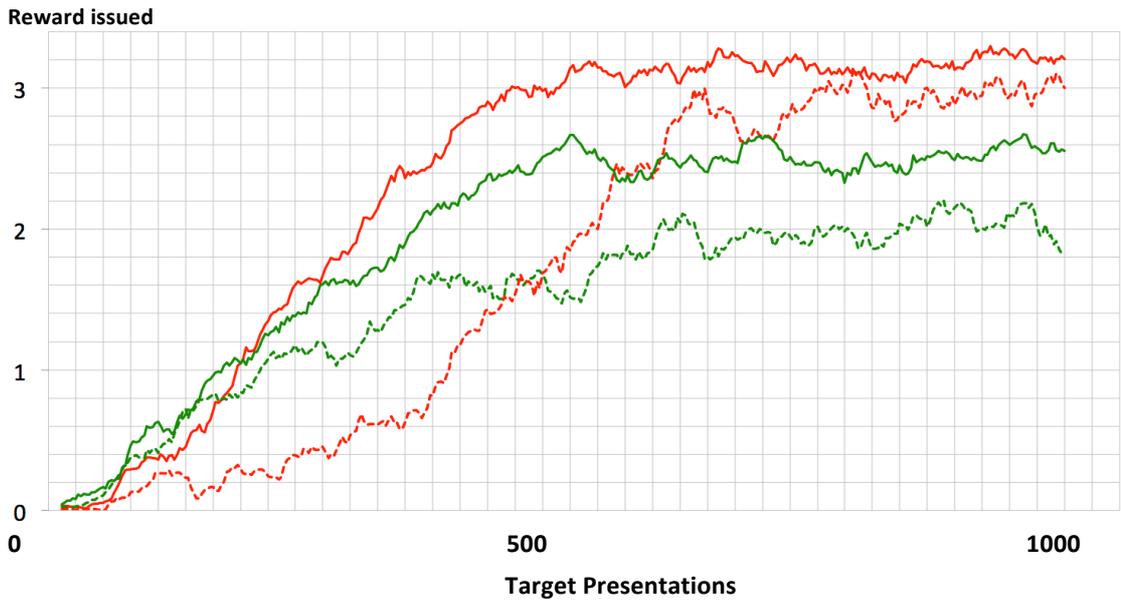
### 5.5.3 Results

Figure 39 compares the performance and learning rate of the synergy-based controller with the original controller by plotting the average reward awarded per trial for both zonal and strike reward types.

Whilst acquisition of both forms of rewards begins earlier for the new synergy-based controller, the most striking change is the increased speed of learning. Most notably, it improves its average strike/touch reward more than 300% faster during the first 300 target presentations and the highest levels reached by the original controller are surpassed by 500 target presentations. The speed of performance improvement in terms of zonal reward is less dramatic (37% faster), however it settles earlier and at a significantly higher level than the original controller.

Overall, it is clear that the synergy controller acquires better performance faster over the same learning period. This can also be readily observed in action, movements are noticeably more dynamic with considerably greater use of turning the torso towards central targets and raising the shoulder to successfully reach much higher targets, identified as a difficult region for the original controller (see Figure 40a for example). A video is strongly recommended to view at <http://tinyurl.com/ECCE-RL3>.

These encouraging results confirm that the candidate synergies uncovered by the factor analysis can be effectively used in a lower dimensional controller that can acquire better performance of the same task significantly faster. However, this also



**Figure 39. Synergy-based controller - average reward by type issued per trial over 1000 target presentations**

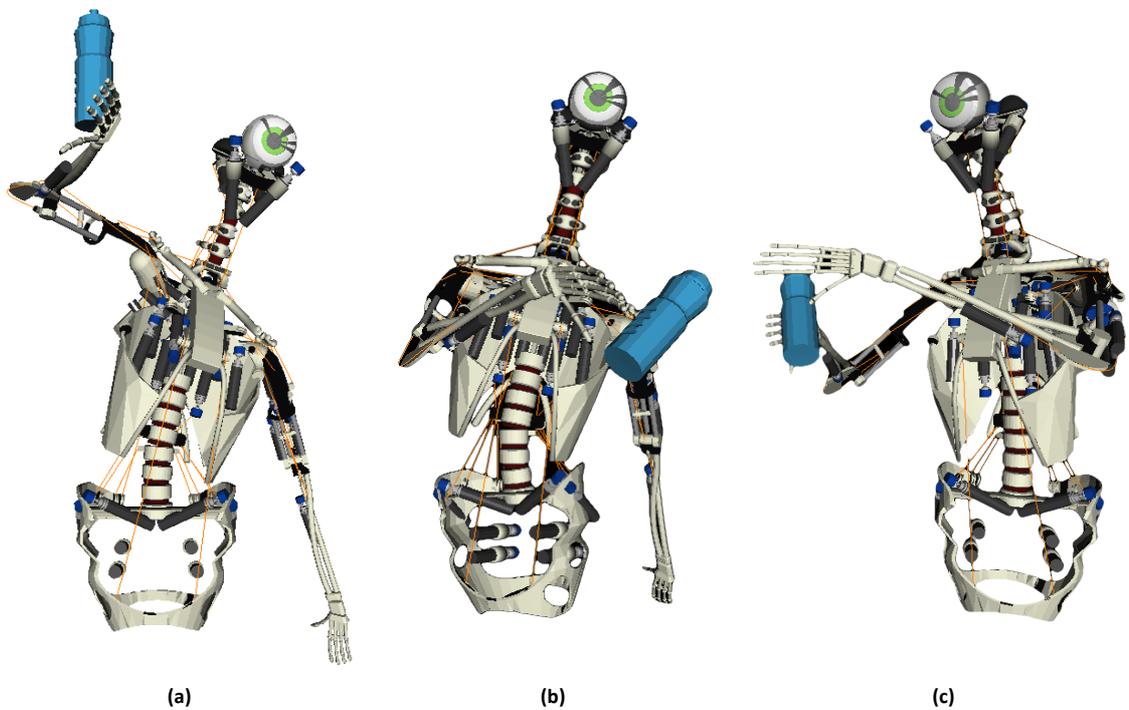
For each type of reward, the 20 trial moving average is graphed, showing the shift in reward awarded over the course of the learning.

GREEN SOLID: Synergy-based Controller – average zonal reward awarded per trial.

GREEN DASHED: As above, but for original controller using individual muscles

RED SOLID: Synergy-based Controller – average strike/touch reward awarded per trial.

RED DASHED: As above, but for original controller using individual muscles



**Figure 40. Synergy-based controller learning the original problem (a) and extended problems (b,c)**

The figure shows screen shots from three different representative learning trials of the synergy based controller.

(a) Reaching with near hand. Synergy-based controller shows improved ability to reach high targets.

(b) Reaching with farther hand. Use of torso rotation increased. Performance degrades with distance from original target zone.

(c) Reaching with both hands. Interestingly, torso rotation is retained to aid reaching-across motion.

raises further questions; are these uncovered synergies linked fundamentally to the structure itself, and can now be applied to the rapid learning of any task, or are they purely task-specific? In the latter case, how many more tasks, and of which kind, must be first assimilated to obtain an “ideal” set that are directly effective in learning any new tasks?

We therefore test the performance of the synergy-based controller in learning two related reaching tasks; reaching across the body and reaching with two simultaneously.

#### **5.5.4 Learning different new tasks using the synergy-based controller**

We undertake the learning of two task variations using the synergy-based controller. Whilst still classified as reaching-based tasks they nevertheless require control of different dynamics triggered in the structure. The learning process and number of trials mirror those performed for the initial reaching tasks, and the aim remains to reach for a target object. The change relates to which hand or hands are rewarded for approaching the target. The performance results for these new tasks over the course of learning are shown as a plot of average reward (strike+zonal) alongside the performance of the synergy-based controller in learning the original “nearest-hand” task (Figure 41a).

##### **5.5.4.1 Reaching to opposite side**

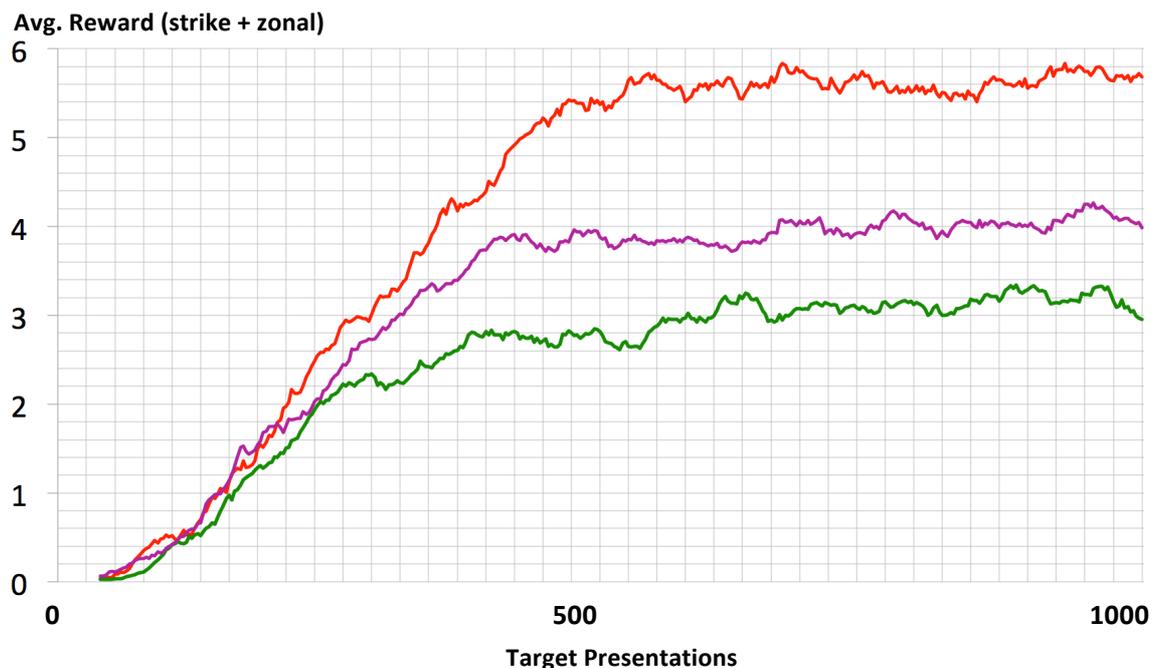
We first consider an incremental change to the task, namely reaching across to a target on the opposite side of the robot. Until now, the nearest hand to the target has been selected, limiting the reaching range required. In this learning trial, reward is issued for movement of the further hand to the target.

See Figure 40b for illustration of the controller in action, or more instructively, it is recommended to view the video at <http://tinyurl.com/ECCE-RL4> . The performance results over the learning trial are shown plotted in Figure 41a (green trace). By comparing with the blue trace we see that the synergy-based controller begins learning to reach across at a similar rate to that it achieved with the nearest-hand task. However, overall performance over the learning falls well short of the original task and it is noticeable from observation (see video) that performance degrades considerably

with the distance of the target from its original learning zone. This suggests that the performance of the synergy set begins drop as the task diverges away from the task used for the synergy analysis. We confirm this by a comparative plot (Figure 41b) of average reward against distance from the original target zone (represented by the horizontal target distance from the mid line).

#### 5.5.4.2 Reaching for a target with both hands using the synergy-based controller

For this task, the robot must now reach for the same target with both hands at once. This task was chosen as a variation for the synergy-based controller since it requires similar arm movements as the previous tasks but considerably alters the dynamics and balance of the body, with both arms moving forward. It also affects the benefit of torso



**Figure 37a. Performance of three reaching tasks learned by synergy-based controller**

For each task, the 20 trial moving average is graphed, showing the shift in average total reward (strike + zonal) awarded per trial over 1000 target presentations.

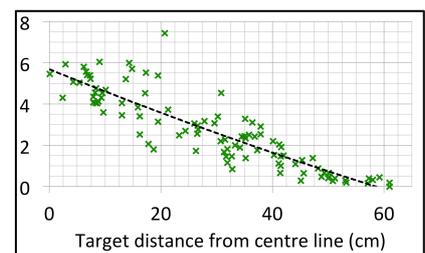
**RED** :Synergy-based controller learning original task (reaching with near hand) used to extract the synergies employed

**GREEN**: Synergy-based controller with task of reaching with the farther hand.

**PURPLE**: Synergy-based controller with task of reaching with both hands.

**Figure 37b: Reward vs. distance from centre line for reaching with farther hand**

After learning to reach with the farther hand the graph shows the variation of reward issued with the distance of the target from centre line, which is used to indicate how far the target is from the region where the synergies employed were learned. We plot the reward (averaged across repetitions) issued for each of the last 100 targets generated in the learning. The trendline (black dashed) is obtained from a 2<sup>nd</sup> order polynomial fitted to the data points.



rotation in reaching, which is clearly leveraged by the single arm controller. Note that, there is a potential for a local optimum “trap” for the learning where one hand only is ever used. To avoid this we create a two-hand-reaching starting set of “rough” actions by merging members of the original initial set used for learning single-handed reaching. Note also that to create a valid reward figure for comparison the rewards scored for each hand are averaged before adding to the stored actions (SARs). We find that the controller proves surprising adaptable to this double-handed task. For example, a notable motor feature to emerge is the continued use of torso rotation for wider targets, but turning in the direction favouring the further, rather than the near hand. See Figure 40c for illustration, or view video at <http://tinyurl.com/ECCE-RL5> ).

From Figure 41a we see that whilst the controller learns this task better than the single handed reaching-across task (purple vs. green traces) the performance nevertheless falls remains considerably short of that achieved by the synergy-based controller for the original task from which the synergies were extracted - namely single-handed reaching to near targets (see Figure 41a - purple vs. red traces).

#### 5.5.4.3 Conclusion

We conclude that the synergy-based controller can be successfully applied to assimilate other reaching-related tasks requiring control of different dynamic forms triggered in the structure. It is particularly notable that, with the large reduction in dimensionality afforded by the synergy-based approach, when the same task is re-learned the reward acquisition begins earlier and at a faster rate than the original controller and reaches a higher level of performance. However, the level of performance drops commensurately with divergence (in terms of target placement) from the original near-hand task. This is particularly noticeable with the single-farther-hand task and implies that the synergy set extracted are not an ideal fit for all movements. We propose therefore that the next logical step would be to modify the learning process whereby the controller may learn a number of tasks simultaneously, with the aim of triggering the emergence of a wider or more flexible range of synergy patterns sufficient to assimilate new tasks without the observed performance degradation (see 5.7.1. *Future Work - Learning core synergies applicable across tasks*).

### 5.5.5 Composition of emergent synergies

We have now obtained evidence that the 13 candidate synergies extracted by factor analysis can be considered artefacts of the dynamic structure and that they can be used effectively unit-wise to control reaching tasks. We now consider what further insights into the role played in movements by the individual emergent synergies may be uncovered by a more detailed examination of their composition. In Table 2 we present a brief discussion of each of the synergies illustrated in Figure 37, looking to identify pertinent features. Study of the table suggest that the synergies fall into three main categories:-

Firstly, there are those synergies (1,3,4,5,6,7,12,13) which can be considered “classical” muscle synergies (i.e. matching the implication of the term as used in biology). Here the activation pattern involves a weighted contraction of a local cluster of muscles with a relatively clear effect in acting on the structure.

Secondly, there are those (2,10,11) that appear to combine two cluster synergies, each acting locally in their own area. These appear to reflect a use for common, proportional activation across disparate body areas and could be potentially considered a hierarchical formation i.e. a synergy of (local) synergies.

Thirdly, there are those (9) that appear to comprise a more disparate group of muscles without clear purpose. These may be an artefact of the analysis, although, more interestingly, it may also be considered an example of a synergy that is effective purely in concert with others, acting in a modulating role. Research into frog synergies has located just such examples; synergies which contribute to no single behaviour but are always found in cooperation with or modulating the outputs of behaviour specific synergies (Bizzi et al. 2008).

A final set to highlight (e.g. synergy no. 8) are those that comprise solely - or primarily - muscle extension (i.e. relaxation) rather than contraction. This is a form that is addressed relatively little in the biological literature, yet can undoubtedly produce effective movement under the force of gravity, such as allowing the hand to drop or to turn the torso using combined tension and relaxation on opposite sides.

Synergy	Discussion
1	This is clearly the primary combination used for the closing of the elbow joint through a mix of the bicep and brachialis. However, the antagonist-like opposing action of the tricep is notable. This synergy emerges - under various weightings - in movements to central, near targets (both high and low).
2	This synergy spans body areas (torso and shoulder) and its use is therefore somewhat obfuscated. However it may reflect an, often simultaneous, need to pull the arm into the body (shortening the pectoralis and latisimus dorsi) and rotate the shoulder inwards (shortening the anterior deltoid while relaxing the posterior deltoid and infraspinatus). This combination emerges heavily in movements to all central targets.
3	This synergy is focused in the back and torso but interestingly spans both sides of the body (reaching and non-reaching sides). It appears to comprise an antagonistic stiffened leaning back of the back but with the rear pulling muscle (lower trapezius) focused on one side, possibly triggering a twisting movement. It emerges in movements to high, near targets.
4	This is focused in the shoulder and scapula muscles and may comprise a combination of rotating the shoulder joint upwards (teres minor) whilst raising the shoulder itself with the trapezium supported by the supraspinatus and lateral deltoid. It emerges to some degree in all movements to high targets.
5	This appears similar to synergy 3 but - rather than leaning back - leaning forward and to the side (towards the target) in a controlled, antagonistic manner. It emerges in movements to wide, far targets.
6	Another back/torso combination, this time nearly balanced left and right. It may cause the torso to lean directly forward, but in a stiff, controlled manner (agonist/antagonist). It appears primarily in movements to distant, central targets.
7	This appears to be the opposite of synergy 1, namely the primary means to extend the elbow joint by shortening the tricep whilst minimally relaxing the brachialis. However, the bicep is also tensioned slightly, suggesting again the antagonist role. Unsurprisingly, It appears to emerge to some degree in movements where the hand has a long distance to travel, requiring a fuller extension of reach.
8	This synergy is distinct as it only contains muscles being lengthened. It appears to be approximately the opposite of synergy 4, allowing the shoulder to drop down. It emerges in various weightings primarily on movements to low targets.
9	This synergy appears to combine a relaxing of arm away from the body with a pulling back of the shoulder. Its utility is not immediately apparent, and appears only in the low, wide, near targets. It may be a result of limiting the number of factors allowable to the factor analysis, alternatively its separation into localised sections may prove more meaningful when applied to other tasks.
10	This appears to mix a clear synergy of the back muscles to allow the torso to lean back with a more incongruous relaxing of the teres minor allowing a downward rotation of the shoulder. This only emerges in movements to low, central, near targets.
11	This is an extensive and very wide-ranging pattern that likely comprises two main disparate elements that are often used together, namely a outward rotation of the shoulder and a rotation of the torso caused by relaxing and tensing opposite sides of the back muscles. The pattern emerges in movements to central targets, bringing the shoulder towards the target.
12	The last two synergies extracted (12 and 13) appear to simply correspond with the two elements that appear in the first co-activation of the two that comprise a motor plan. The first is a simply synergy that raises the shoulder, the natural body dynamics cause it to swing forward, initiating a reaching movement. It appears in greater weightings according to the distance of the target.
13	The second synergy element causes a simply pulling back of the opposite arm, we assume this provides a beneficial counterbalancing effect. Again, its weighting appears to correlate with the horizontal distance of the target.

Table 2. Detail and discussion of synergy composition

## **5.6 Discussion and potential implications of the findings**

### **5.6.1 Introduction**

In this section we will discuss a number of key aspects arising from this set of experiments testing a reaching controller design based on muscle co-activations and synergies emergent through elementary reinforcement learning exploiting amenable dynamics in the biomimetic structure. Firstly, we discuss the potential for transfer of this learning approach to the real robot. Next, we will discuss briefly any possible biological implications of our findings. Finally, we will outline what we consider to be the limitations of the study, leading into the next section, which will flesh out a number proposals for potential future work.

### **5.6.2 Transfer of approach to physical robot**

An major underlying purpose of this investigation has been to locate potentially effective approaches to control a physical biomimetic, musculoskeletal robot, in this case, the anthropomimetic ECCERobot. Do results suggest that our approach fulfils this brief, what further work is needed for transfer from the model and how would the learning and experiments be adapted? Two potential avenues offer themselves:-

The first, which applies to other control techniques also, is to create a sufficiently precise and comprehensive physics model of the robot such that motor plans learned on it are directly transferrable to the physical domain. This would provide scope for the use of any and all computing resources to be directed offline at the problem – such as hundreds of trials, hours of simulated learning or GPU acceleration. The problem here, of course, is the creation of such an accurate model. Our model, whilst comprehensive and comparably complex in its own right cannot claim to be more than a fair approximation of the true robot. Techniques such as correction of model parameters via hill-climbing error correction from simultaneous model/robot trials (e.g. Wolpert & Kawato 1998; Haruno et al. 2001; Wittmeier et al. 2012) holds some promise of improvement but cannot truly incorporate, for example, the breadth of materials, joining techniques and complex friction that feature in the physical robot.

An extension of the model based approach would be to employ an imprecise model to generate approximate plans but use feedback control to correct. This is a well trodden path in control techniques (Franklin et al. 2002) but requires an extended controller to generate and manage feedback signals and a sufficiently fast sensorimotor loop to

avoid instability (Smith 1959; Miall et al. 1993; Franklin et al. 2002). However, these issues are compounded by the presence of a highly dynamic and non-linear control subject, such as a anthropomorphic robot. They can be combated with techniques such as incorporating prediction into the feedback signal (Smith 1959), but this in turn requires a model, thus the problem risks becoming tautological. Furthermore, the current model runs at only a fraction of the speed required to avoid significant delay issues for live, real time control. This would necessitate significant delay compensation if the model were to be incorporated in a physical robot controller. A better alternative solution may be to employ the longer feedback cycle of *model predictive control* (Kwon et al. 1982; García et al. 1989; Mayne & Michalska 1990; Cueli & Bordons 2008) which additionally incorporates prediction of the environment state, critical for a robot placed into the real world. A proposed continuous controller architecture for the physical robot incorporating delay compensation and based on model predictive control (MPC) is derived and trialled in Chapter 6.

The second potential avenue to pursue is to employ learning and analysis techniques that can be applied directly to the robot itself, or at least refined “in-vivo” after using a model to kick-start learning if numerous learning iterations to be expected. This approach also has the potential advantage, in common with animal systems, of being able to adapt to “plant drift” (e.g. dynamic changes due to wear and tear) as biological bodies do also. Whilst such a model-free approach will eliminate many control approach candidates from the field, we suggest our approach is not among these. In theory at least, the experiment undertaken here used here could, in fact, be directly repeated to teach target reaching to the real robot, with two provisos.

Firstly, a means to capture the hand trajectory in space is required (a fixed external Kinect sensor could provide this) and secondly, that the number of trials needed must be reduced sufficiently to become a realistic proposition on a high powered and relatively delicate structure. In the future work section (5.7) we therefore discuss possible means to significantly reduce the number of learning trials by capturing more information from those performed.

### **5.6.3 Biological Implications**

We consider briefly any potential implications for theories of biological motor control arising from our findings.

### ***5.6.3.1 Findings support claims of fixed pattern motor synergy theories***

This research found that a very complex compliant structure that challenges conventional control approaches could be controlled to perform effective directed reaching actions simply through the application of linear combinations of correctly weighted muscle co-activations under simple driving signals. This practical demonstration provides considerable further support for theories that postulate that this form of approach is employed widely in nature, from frogs, to cats to humans, where muscle signal analysis has previously suggested this to be the case (Ting & McKay 2007; Verrel et al. 2010; D'Avella et al. 2003; Cheung et al. 2009; Cheung et al. 2005; D'Avella & Tresch 2002; Tresch & Jarc 2009; Ma & Feldman 1995; Hart & Giszter 2010; Bizzi et al. 2008). Using the same analysis techniques to propose candidate synergies from general muscle co-activations learned under reinforcement learning, we find evidence that these can be employed directly as units in a low dimensional controller to learn the same reaching task faster and better and - to a reasonable extent - applied to related tasks requiring control of altered dynamics. By fitting the synergy set to co-activations learned in alternative runs, we also find evidence suggesting that these candidate synergies reflect dynamics within the structure rather than comprising artefacts of the course taken by a specific learning trial. In the future work section (5.7) we therefore propose comparison with synergies extracted from this structure through "balanced truncation" (Berniker et al. 2009) and potential adaptations to our approach facilitating the extraction of a more generic set of synergies applicable to a wider range of motor tasks.

### ***5.6.3.2 Very simple reinforcement learning is sufficient to uncover effective weighting patterns and driving signals to provide elementary reaching control***

Whilst the neural correlates of some components of reinforcement learning methods appear to have been identified (such as dopamine release), the mechanisms that could locate the complex muscle signals though to be needed to control the highly dynamic compliant body have been far less obvious. However, as demonstrated by these results, the muscle synergy approach lends itself far better to simpler learning mechanisms, the implementation of which by the brain appears far more plausible and the neural correlates of which may perhaps be more easily identifiable.

### ***5.6.3.3 Role of Compliance***

By comparing the reaching performance attained through learning on a model with significant compliance in the muscles to the performance of one with negligible compliance (by altering the modelling of the shockcord employed in the robot muscle cables), we observe a distinct contribution to the reduction of jerk when switching motor activation patterns. We conclude that, while detailed analysis of forces and energy has not been undertaken, the evidence from force measurements suggests that the control emergent under RL was able to exploit motor cable (“muscle”) compliance as a spring energy store to reduce the force required when switching between stages of a movement, resulting in more reliable outcomes through lowering of signal-related noise, and aiding smoother (lower jerk) movement.

### ***5.6.3.4 Optimality effects of reinforcement learning on musculoskeletal, compliant structures***

The experimental results support the theory that reinforcement learning will favour reliable actions when trials are repeated (Wolpert et al. 2001). After adding signal-dependent Gaussian noise, (but not for fixed-level Gaussian noise) then, as predicted by the optimality theories put forward by Harris and Wolpert (1998), for those target regions where the robot has learned to significantly slow the hand on arrival we do observe a migration towards the stereotype bell-curve “signature of optimality” velocity profile, as well an increasing smoothness of movement (lowered jerk) and reliability (lowered endpoint variance) across all target regions. However, it should be noted that these findings were solely based upon explicitly repeating (almost) identical problem states whilst varying only the noise, a somewhat unrealistic proposition in nature.

### ***5.6.3.5 Hierarchical synergies and implications for motor learning and cognition***

EMG evidence from biology suggests that synergies are locally clustered (d’ Avella et al. 2006; Cheung et al. 2009) but can coordinate with clusters from other body areas (Ma & Feldman 1995). This suggests a more hierarchical layout of synergies, with a higher level comprising, in effect, synergies of (local cluster) synergies.

In our learning control muscle activations were allowed to include muscles across the whole body, therefore one activation might include use of both the arm and the torso muscles, even the opposite arm. However, under factor analysis, reuse of smaller fixed

patterns emerges, most of these are clustered within a local area of the body, forming an clear analogy to the biological synergies.

However, we also see patterns that appear to combine two local cluster synergies, each from a different body area. These appear to reflect the utility of proportional activation across disparate body areas - these could therefore be considered a emergent hierarchical formation i.e. a synergy of (local) synergies. Such a hierarchical or layered model may serve to explain why seemingly contradictory evidence exists that synergies are learned (Ting & McKay 2007) against both anatomical evidence (Li et al. 2008) and EMG analysis (Cheung et al. 2009) which suggest that they are hard-wired at a lower level. It may be that local cluster synergies are pre-wired but are themselves synergised together into prefixed activations through motor learning. This pattern may even repeat at one or more higher levels allowing richer behaviours to become autonomous and subconscious, as is also proposed by Ting and McKay (2007).

#### *5.6.3.6 Could muscle-based reaching control in the brain resemble our approach?*

Finally, without attempting to be specific about anatomical detail or neural correlates, we make a brief speculative claim that the principles of our strategy are arguably applicable to the fast planning in the brain - followed by commencement - of a motor behaviour such as reaching. Two elements are key to this claim. Firstly, that simply sustained activation of the correct pattern of muscles under a simple template-based driving signal, is sufficient to produce effective actions on a structure that has co-evolved to offer amenable dynamics to such relatively simple control signals. Secondly, that the vector summation of the selected muscle synergies that is required to obtain the final output set of individual muscle contractions, could, in theory, be achieved directly at the motor neuron cluster outputs rather than requiring the neural implementation of some form of intermediate summing unit.

Under this approach, when a problem or task is presented, the synergy combinations that triggered past movements re-activate as clusters through sensorimotor associations developed from a combination of Hebbian-style association (Hebb 1950; Bienenstock et al. 1982) and spike timing dependent plasticity (STDP) strengthened through reward-triggered dopamine release (Izhikevich & Desai 2002; Schultz 1998). Due in particular to the simple sustained activation of these clusters, these associations could emerge relatively straightforwardly, without the need for neural correlates of

the more complex learning mechanisms of distal reward, eligibility traces or temporal difference that have been designed to enable machine learning of successful sequences of micro-actions.

With the strengthening of these associations, the proximity of the new problem state to those encountered in the past can be expected to trigger a greater or smaller response within these synergy clusters, which then activate the motor neuron clusters of the relevant individual muscles. These clusters are simultaneously activated by other synergy groupings, each similarly weighted by this proximity. These produce a net activation on each muscle that correlates to the weighted summation of the original synergy patterns, i.e. weighted by their strength of association with the task.

#### **5.6.4 Limitations of the study**

In this section we will attempt to highlight shortcomings we have noted in the method, results or conclusions claimed and how they might be defended or addressed. Some of these lead directly into proposals for future work which form the next section of this discussion.

##### ***5.6.4.1 Not proven as general controller - only shown for specific reaching scenarios***

Whilst the study reveals the potential of motor synergies in simple combination to control very complex, compliant biomimetic structures and the possibilities of employing reinforcement learning to uncover them, the problem scenarios employed are considerably simplified from any real world situation. For example work to date has been limited to trialling reaching from an almost constant single start state.

This throws doubt on claims that this approach, in particular the learning aspect, is applicable to progressively more complex scenarios, up to the ultimate control scenario where the subject can perform any given task from any starting dynamic state. The *Further Work* section below therefore proposes studies that extend the problem state significantly.

Similarly, the controller is unproven for learning of movements calling for 3 or more chained co-activations, nor has it demonstrated the ability to encompass follow-on tasks such as grasp object, set down etc.

#### ***5.6.4.2 Bio-mimetic claims are unproven***

An important claim made is that the specifically biomimetic nature of the model providing exploitable amenable natural dynamics leads to the success of this surprisingly simple synergy combination approach. However, it cannot currently be fully discounted that any musculoskeletal structure, bio-mimetic or not, could prove controllable by the correct set of synergies. This claim would therefore be considerably strengthened if the same approach were tested against complex, but non-biomimetic structures, perhaps randomly generated.

#### ***5.6.4.3 Candidate synergies uncovered by analysis not proven as superior to a random set***

Claims of the emergence of effective, combinable synergies would be strengthened by testing a null hypothesis that a randomly generated set of 13 (possibly linearly independent) candidate synergies performs as well in a controller as the set uncovered by factor analysis following a period of task learning using co-activations of individual muscles.

#### ***5.6.4.4 Biology insights are limited by nature of controller***

Although the model is bio-mimetic the controller essentially comprises an algorithm rather than a biomimetic brain (such as an extensive spiking neuron simulation) and therefore claims of insight into biological control must be treated with care. However, even biomimetic brain simulations, whilst providing potentially valuable insight, can also be easily argued to lack true verisimilitude in any depth. Thus points made with regard to control principles from algorithmic evidence remain as similarly valid and interesting as those arising from many spiking network models so long as their limitations are acknowledged.

#### ***5.6.4.5 Alternative method for uncovering key synergies not pursued for comparison***

The approach of “balanced truncation” analyses the structure dynamics directly to extract a low dimensional model for use in key synergy identification (Berniker et al. 2009). Its findings suggest support for the general claims of this study regarding synergy control and remains an interesting and potentially applicable method for this model structure. A comparison of the outputs of this analysis with the results of our learning would therefore have formed a useful check on the claims made here, and

provided useful insight into the advantage and disadvantages of either method. Such an extension is proposed in the *Future Work* section.

#### ***5.6.4.6 Unproven against physical robot***

The findings are limited to the model, whereas a demonstrable means to control the real anthropomorphic ECCERobot would be of significantly more interest. In practice the approach might prove to be too slow with too many repetitions, or the model may prove too different from the robot to be able to carry across findings.

#### ***5.6.4.7 Approach lends itself to overly homogenous movement profiles***

Observation of the reaching in action shows immediately that any given target location is addressed with a movement bearing a strong general resemblance of form to many others. This homogenous nature is likely a direct result of the plan-merging approach which quite plausibly acts to rapidly dilute distinctive movement and leads to all tasks being addressed with a generic style of movement that may not be the best for each task. For example, a target location directly in front of the breastbone is responded to by a somewhat wasteful and extravagant hand movement that sweeps in from the side, when a simple raising of the arm to the front and centre would appear more appropriate. The extraction and subsequent freezing of synergy patterns used in these movements will almost certainly act to exacerbate this situation.

However, in partial mitigation it should be recalled that human movements are very often not straight and the underlying reasons remain unclear (Petreska & Billard 2009). The use of generic synergies may even suggest a possible reason.

#### ***5.6.4.8 Optimality/reliability investigation minimal – claims may be extravagant***

The study looks to draw some relatively important conclusions on the role of reinforcement learning and reliability in generating movements that display signatures of optimal control. However, whilst bell-curve profiles and reliability gains were noted, the precise causes remain explored in relatively little depth and null hypotheses are not investigated beyond reasonable doubt (for example investigating learning outcomes without trial repetition).

## 5.7 Future Work

In this section we will discuss potential further work that we believe would prove rewarding to undertake or that may address criticisms raised in the previous section.

### 5.7.1 Learning core synergies applicable across tasks

We have tested the idea that a set of core synergies, once identified, can be used as building blocks to rapidly develop actions targeting new, but as yet unattempted, tasks. However, we found that although the synergy-based controller could rapidly assimilate some other related tasks, the level of performance dropped with divergence from the original simple task (from which the synergies were identified), in particular with regard to target placement. Other techniques such as “balanced truncation” (Berniker et al. 2009) are less task-related through their direct extraction from the underlying structure dynamics themselves. In our case it appears that the synergies emerging are not sufficiently generic for other roles due to the limited arena of the task they were learnt on.

Although this can be argued to be a recognisable facet in nature also - entrenched motor habits are notoriously difficult to unlearn (Brashers-Krug et al. 1996) - to control the ECCERobot we would ideally wish create a better, more task-generic, controller through an extension of the methods we have so far applied. This suggests undertaking learning with a range of tasks with the aim of extracting a more flexible and powerful set of synergies at the end. Three possible approaches are suggested here.

Firstly, to use the original controller to separately learn each of a number of tasks, then to analyse as before, but across all the learned co-activations, for a set of candidate synergies for trial in a synergy-based controller.

Secondly, to learn more slowly a range of tasks all at the same time, in effect, setting the task category as a dimension of the problem space. Task selection could be random but based on a distribution favouring importance and common usage. In implementation terms this could equate to attaching not one but a set of rewards to each plan, each referring to its success at a different task. However, the bank pruning would be applied to the plans with the least total reward, with the intention of eliminating specialisation and encouraging generalisation across tasks.

The final proposal is to learn only a single task that is nevertheless general enough to subsume the requirements of most of the others. For example, this could involve reaching to two different targets using both hands, thus subsuming all the other tasks attempted to date. This requires no modification of the reward mechanism but will be slower to learn, as it doubles the problem space to 6 dimensions covering the  $[x,y,z]$  location of both targets.

### **5.7.2 Incorporation into general control architecture based on MPC**

As discussed earlier when considering the potential transfer to the physical robot, in the next chapter we propose a general delay-compensating control architecture for the physical robot based on Model Predictive Control (MPC). A logical part of that framework will be the potential incorporation of the synergy-based control approach as the realisation of the envisaged generic planner module. This will be discussed at the end of that chapter after the architecture has been introduced.

### **5.7.3 Extending the problem space - commencing from any state**

As discussed in the limitations of the study (see 5.6.4) one reason that we cannot claim to have produced a generic controller is that work to date has been limited to trialling reaching from an almost constant single start state. As argued, this simplification was intended to aid validation or otherwise of the principles of the synergy and co-activation approach by minimising the input problem state to the target position alone. Nevertheless, to be of genuine use in controlling a robot such as ECCERobot we must consider how we might learn to reach a target from any necessary dimensionality of problem state. For example, Table 3 below shows how the problem state might extend to cover a much wider range of initial robot states. The crucial point to halt this dimensionality growth is likely to fall where enough of the state is captured for the controller to function effectively in a control loop where a new action can be continually reissued as the state is periodically captured on sensors, or at least estimated. An example of such a control system is covered in more detail in the next chapter.

It can be seen from Table 3 that we can rapidly reach 21 dimensions in the problem state without even beginning to consider elements such as joint angles (posture). For the current approach to function in higher dimensions would therefore require a richer state estimator (Sutton & Barto 1998) to be developed, and to weight the influence of

different dimensions according to their impact on the problem. Without this, using only a linear nearest-point measure, the closest plan to the problem state will disproportionately dominate weightings since the other plans are simply too distant when so many dimensions are in play. It may be possible to optimally balance these weightings dynamically during learning, as with an actor-critic RL approach (Sutton & Barto 1998) or to use the balanced truncation analysis discussed earlier (Berniker et al. 2009) to ascertain those dimensions with the most influence (see 5.7.5.1).

However, there are also other potentially interesting approaches to managing problem state dimensionality, resembling the way that the current controller improves its behaviour through the rewarding of reliability.

For example, a simple experiment would be to extend the problem to that of reaching to a succession of targets *without any state reset* between trials. This shifts the learning focus from solving the raw dimensionality of the problem state to locating strategies

PROBLEM STATE	ADDED DIMENSIONS	TOTAL DIMENSIONS
Standard starting robot state at rest + target position	-	3 (target xyz)
Nearest hand position, velocity discounted	3	6
Nearest hand velocity vector	3	9
As above , for both hands	6	15
centre of gravity of whole torso, relative to the base	3	18
Add velocity vector for centre of gravity	3	21

**Table 3. Extending dimensions of the problem state**

where the previous actions can take a direct role in maintaining a sufficiently low dimensional (i.e. simpler) problem for the upcoming one. Thus, if the problem state comprises only the hand and target positions, then more “cautious” reaching where the hand and body finish in relatively stable states will reap benefits in the next reaching task, whereas creating highly dynamic states by overly forceful movements of the hand and body at targets often may not. Furthermore, apart from just reducing the problem state dimensionality we are likely much nearer to the real situation for a human or physical robot, where continual accurate state resetting is unrealistic or at least problematical.

For a first experiment in extending the problem state we recommend a continuous (i.e. no reset) trial of the controller extending the problem state to 9 dimensions by adding the position and velocity vector of the nearest hand to the target. This is a relatively

low extension of problem state but brings the advantage that the controller could potentially now be employed in a continual re-plan mode where a new action is iteratively generated for the current state of the hand as it approaches the target. As discussed, a delay-compensating version of such a feedback controller is detailed in the next chapter. A particular point of interest would be to look for the emergence of synergies acting to stabilise the hand or torso dynamics, thus reducing the real problem state that affects the success of an action towards the low-dimensional version of the problem state that we are setting the controller to solve.

#### **5.7.4 Trajectory storage approach for speeding of learning for physical robot**

Although the ability to apply our control approach to a physical robot is an important goal, we face the serious issue that even with the faster-learning synergy-based controller a significant number of trials with the model are still currently required to reach an acceptable level of performance. It is likely that the high power requirements, wear and tear and current fragility of the robot make this shift potentially unrealistic at present. Furthermore, using the synergy-based controller raises the issue that, unless the physics-based model can be aligned considerably better to the real robot, it is debatable whether synergies extracted from training the model will be sufficiently effective for the real robot. We would then face the prospect of training the original individual-muscle controller on the real robot, an even longer process.

To attempt to address this, we note that a trial of an action does not only provide a potential route for the hand to the specified endpoint, it also gives a route to every point along the trajectory. To date, we discard all these in favour of one single point we consider the “best” suited endpoint to represent this action. Yet, at least during earlier stages of learning, the action offers a reasonable attempt to reach all the other points along its route. Recall that the RL approach is often more effective at improving and optimising poor actions, and less effective at locating useable actions from scratch. It therefore makes sense at the early stage of learning to bootstrap as much information from reasonable trial movements as possible.

We would therefore propose an extension where each stored action holds not simply a “best” target location but a mapping of the hand trajectory and its velocity profile. When a new target is presented each stored action puts forward the best waypoint it

can offer for that target. These are then weighted, as before, against those offered by other stored actions.

Early trials of this approach are promising, suggesting that significantly greater reward begins to be issued at an earlier stage in the learning. This may therefore prove a fruitful route to transfer of the control to the physical robot.

### **5.7.5 Comparative studies with alternative methods**

This potential future work would seek to test alternative learning methods developed for synergy identification and synergy-based control. By applying these to the same model and task, informative comparisons may be made regarding the validity of our findings and conclusions, the nature of synergies located, the speed of learning and its effectiveness.

#### ***5.7.5.1 Comparative study with low dimensional model extraction with matched optimum synergies***

As discussed in the *Limitations* section, it has been stressed that the trigger for developing this approach was the biomimetic nature of the robot. However, it has not been shown definitively that it is specifically the biomimetic nature that creates amenability to synergy-based control, and it may be simply that biomimetic structures fall within the set that possess key features that make them amenable. Applying analysis techniques such as the low dimensional model extraction developed by (Berniker et al. 2009) against this model and other non-biomimetic musculoskeletal structures may reveal the key differences as well as providing a useful check for the validity of our findings and conclusions and comparison of the synergy patterns located.

#### ***5.7.5.2 Synergy emergence from core reflexes via mutual information and natural dynamics***

As discussed in the *Background* chapter (see section 2.4.3) this branch of morphological computation (Der 1999; Te Boekhorst et al. 1999; Lungarella & Sporns 2006; Pfeifer et al. 2007) seeks gradual emergence of synergy-based control through implantation of a minimum set of core reflexes combined with mutual information association (e.g. Hebbian learning) formed by correlation between co-occurring proprioceptive and motor signals (Gravato Marques et al. 2013). Current work in this area is heavily focused on direct application to physical, but significantly simpler, biomimetic robots

rather than software-based models. It seeks to eventually master the full complexity of a anthropomimetic robot such as ECCERobot via a series of increasingly complex structures.

### 5.7.6 Creating hybrid, synergy-based control approaches

We would propose revisiting several control approaches considered in the *Background* chapter and consider their suitability to create a hybrid approach based on exploiting the lower dimensional control offered by using analysis-extracted, synergy-based control units rather than individual muscles. With this significant reduction in dimensionality, we may now be able exploit these generic, proven and powerful control approaches to create general controllers for anthropomimetic robots such as ECCERobot. These approaches might therefore include; generic and high-dimensional reinforcement learning, high-dimensional planning search (Shkolnik & Tedrake 2009; Kavraki et al. 1996; Kavraki 2007; Rusu et al. 2009; Ladd & Kavraki 2004; Kagami et al. 2003), evolutionary and artificial neural network robotics (Beer 1995; Cliff et al. 1993; Meyer et al. 1998; Bongard 2009) and control through equilibrium point (EP) hypothesis (Gu & Ballard 2006b; Feldman et al. 1998).

## 5.8 Conclusion

Whilst by no means a complete control solution we suggest that the results of these experiments strongly indicate a fruitful area of investigation for control of bio-inspired, biomimetic structures such as the anthropomimetic robot ECCERobot.

The complex and compliant modelled control subject was not created with any compromise for ease of engineering control and undoubtedly comprises a highly challenging control task for well-established approaches such as classical control, planning search, generic reinforcement learning and evolutionary/neural network robotics.

However, in apparent contradiction of this complexity, there is strong empirical evidence that combinations of surprisingly simple sustained signals driving a common set of muscle activation patterns in weighted proportions have been shown to underlie controlled movement in frogs, cats and humans. Correspondingly, we show here that control of this complex biomimetic model is also demonstrably susceptible to this synergy-based approach to dimension-reduction when it is applied using simple

reinforcement learning, an originally bio-inspired trial-and-error technique that can leverage amenable natural dynamics. Furthermore, whilst most biological studies reveal evidence supporting the theory that synergies may be employed unit-wise by these animals, here we demonstrate that the form of extracted muscle activation patterns proposed as synergies by these biological studies can indeed be explicitly used in this way to form an effective reaching controller for a complex biomimetic model.

The newer “brute force” techniques we have reviewed such as generic high dimensional reinforcement learning and GPU-accelerated planning search do not employ the biomimetic nature of the musculoskeletal structure to their advantage and would therefore still face, in the case of this control subject, a potentially insurmountable curse of dimensionality. Only specific musculoskeletal techniques such as equilibrium point hypothesis and alternative synergy-based approaches such as low-dimensional model extraction (Berniker et al. 2009) or reflex-based emergence based on mutual information theory (Gravato Marques et al. 2013; Wittmeier et al. 2013) potentially offer equivalent promise.

For biomimetic structures such as our model, we have shown evidence that reinforcement learning has acted as an action discovery mechanism, uncovering some simplifications of a solution through exploiting amenable natural dynamics of the biomimetic structure, which were not apparent or designed by the robot engineers.

We have demonstrated that we can apply optimality principles to encourage the emergence of smoother movement by incorporating both signal dependent noise and trial repetition into the learning process. Across all targets regions we observed an increasing smoothness of movement (reduction in jerk) and an increase in reliability. Furthermore, as predicted by Harris and Wolpert (1998), these results applied when adding signal-dependent Gaussian noise, but not for fixed-level Gaussian noise. We also found that, for those target regions where the controller has learned to significantly slow the robot’s hand on arrival, we do observe under reinforcement learning some migration towards the stereotype bell-curve “signature of optimality” velocity profile.

Compliance is a primary feature that sets both biological bodies and these musculoskeletal robots apart from conventional stiff-jointed robots, but adds significant complexity when employing conventional control approaches. By

comparing learned control of compliant and non-compliant models, we investigated the effects of compliance on our approach. Initial results suggested that the compliance in our model contributes to a reduction in jerk, thereby smoothing movement, and furthermore, acting as an energy store allowing for a reduction in the motor force needed for direction changes, resulting in a drop in the signal-related noise that causes unreliability.

Finally, we conclude that the most compelling areas to take this work forward are firstly, adapting the learning process to reveal more “generic” synergies applicable across a wider task range. Secondly, to adapt and extend the controller to transfer it to control of the real robot. Once such extension, the adoption of a general control architecture for continuous control, is explored in Chapter 6.

## Chapter 6

# A bio-inspired continuous control architecture for an anthropomorphic robot incorporating environment integration and delay-compensation

---

### 6.1 Introduction

The previous chapters have focused on addressing some specific control issues for a musculoskeletal robot when tasked with a closely defined reaching task. However, a robot controller for the real world also requires an overarching control architecture if the robot is to remain under continual control for a period of minutes or hours performing a series of tasks in a dynamic environment. This chapter discusses and proposes a design of a continuous control architecture potentially suitable for compliant anthropomorphic robots such as the ECCERobot.

An important early decision in choosing a robotic controller is whether to follow a model-based design. Whilst certainly a dominant approach in control engineering, there is also considerable evidence from neurobiology that predictive modelling is employed in the human CNS, addressing control issues such as noise, state estimation, delays, motor planning and integrating the sampling and prediction of external variables (the environment). For example, use of predictive mechanisms provides a highly plausible reason for the clear physical presence of motor efferent copy nerves (Sperry 1950; Kelso 1977). It has also been shown that faster reaching movements are planned and performed too quickly for any feedback-led mechanism to be driving them (Desmurget & Grafton 2000). The existence of prediction-based delay compensation is also supported by studies suggesting that the self-perceived “current state” employed for planning a motor task may not comprise the state captured at the moment of sensory input but rather a prediction. For example, Ariff et al (2002) found that the position of eye saccades tracking an unseen reaching movement appeared to reflect the output of a state predictor, rather than the actual position of the hand after it had been subjected (unknowingly) to a force field.

The assumption of Kalman filter-like predictive mechanisms being used for accurate state estimation (Balakrishnan 1978; Welch & Bishop 2006) by the CNS also predicts several unusual observed phenomena such as the cutaneous rabbit illusion (Kilgard & Merzenich 1995; Grush 2004) - where a series of taps on the arm appear (wrongly) to the subject to have followed a smooth extrapolated path - and also the auditory continuity and phonemic restoration illusions (Grossberg 1995; Grossberg & Myers 2000), where interruptions in sensory data are not perceived at all by a subject so long as it swiftly resumes along a predictable path. Wolpert et al (1995) also found that end point estimation data following reaching movements made in the dark was best accounted for by a Kalman filter model combining an internal forward model with sensory correction.

Nevertheless, non-model motor controller designs such as behaviour-based robotics (Brooks 1991b) have enjoyed success and the need for - and evidence of - predictive models within neurobiological motor systems remains a topic of debate, particularly in animals with relatively simple cognitive abilities (Brooks 1991; Miall & Wolpert 1996; Webb 2004).

However, as we have already developed a rich physics-based model of the robot, we look to leverage this work by drawing upon established designs from control engineering that do employ predictive forward models to address control challenges such as noise, state estimation, delays, motor planning and integrating the sampling and prediction of external variables (the environment).

The problem we face is that, given the complexity of the robot, the lack of mathematically tractable model and the presence of a potentially rich dynamic environment, there is no single control engineering design that can be applied as-is to this problem. We therefore look to a combination of modules and technologies to form a novel proposed controller for this family of robots. However, our design draws most from the well established *model predictive control* (Kwon et al. 1982; García et al. 1989; Mayne & Michalska 1990; Cueli & Bordons 2008). This approach can be thought of as an extrapolation of prediction-based feedback control methods - such as the Smith Predictor (Smith 1959; Franklin et al. 2002) - but extended to encompass the predicted future of the *complete system under control and its environment* in a more extended iterative cycle.

In particular, we look to fulfil the delay-compensation features of this approach through the use of the physics-based model of the ECCERobot (see Chapter 3) as a predictive component, and we demonstrate its effects whilst attempting to control a second copy of the model acting as a proxy for the real robot. We show that performance is indeed significantly improved if a precise degree of delay compensation is applied and additionally that the performance of an uncompensated controller falls off steeply if modelling accuracy is less than perfect (as would certainly be the case when employing the real robot).

Finally, we consider an interesting potential implication of these findings for our own conscious perception of “now”.

## **6.2 Issues Arising In the Design of Robot Controllers**

Although a rudimentary ideal controller can be easily designed, a number of potentially serious issues swiftly arise one when implementation is attempted for a real robot.

Firstly, capturing not only the robot’s own state, but sufficient of the (potentially rich and dynamic) environment state and the relative position of one to the other will be critical to success, yet this is not a trivial task. Secondly, what form of sampled state, and in how many dimensions, is required for a motor planner to function effectively for a given robot? Thirdly, real sensor data will be affected by noise and inaccuracy, translating directly to a misreading of the state. Lastly, in all systems, there is an unavoidable delay between taking a sensor reading and the ensuing re-planned motor signals finally reaching physical actuators.

If this delay is too large or the robot/environment states are too dynamic then the state used to plan from will be significantly out of date resulting in control errors. This affects both open-loop and closed-loop (feedback) controllers. Open loop controllers must generate accurate motor plans, requiring more planning time and model simulation time, during which the robot and environment states will have changed from that sampled. Feedback controllers can issue more rudimentary plans and then correct via a state error signal but this requires a very fast sensorimotor loop, resampling the robot state to generate the error signal, else the delay will distort the error, leading to instability (Miall et al. 1993).

The ECCERobot itself presents a particular challenge as a control subject, having limited proprioceptive sensor range (primarily relatively poor muscle tension sensors) combined with a high-dimensional non-linear, elastic structure making its state both hard to predict and fast changing.

### **6.3 Principles of model predictive control (MPC)**

Model predictive control (Kwon et al. 1982; García et al. 1989; Mayne & Michalska 1990; Cueli & Bordons 2008) can be considered an extrapolation of prediction-based feedback control methods such as the Smith Predictor (Smith 1959; Franklin et al. 2002). These feedback methods use a forward model of the system to generate a fast estimate of the state from the motor signals sent out, without waiting for the real state to be resampled, thus avoiding potential instability. This estimate is then fed back to provide an error signal to drive correction motor signals.

In MPC this principle is extended in both scope and time to encompass the predicted future of the *complete system under control and its environment* in an extended iterative cycle. It can thus be considered a form of closed loop feedback controller but operating over longer iterations, implying a greater need for the kind of planning and modeling accuracy usually seen in open loop controllers.

Importantly, MPC incorporates the integrated sampling and predicting (modelling) of independent dynamic variables (the robot's environment in this case) and implements a continual cycle comprising longer iterations that commence with a sample and prediction of state up to a fixed "horizon" point in time which can be set to allow for all delays inherent in the system (sensing, simulation time, planning time etc). Control plans are then repeatedly revised, based upon this predicted future state. For this reason MPC is also referred to as a *receding horizon* approach (Kwon et al. 1982; Mayne & Michalska 1990).

MPC is attractive for us as we have a complex non-linear system with poor proprioception approximately modeled by a slow physics-based simulation which is nevertheless potentially equipped to incorporate a dynamic model of the environment, if this can be effectively sampled.

## 6.4 Use of the ECCERobot Physics-Based Model

By incorporating a model of the robot that can be updated from raw sensor readings we can decouple the motor planner from the potentially limited, noisy and inaccurate set of physical sensor readings. For example, if the planner requires velocity data, then rather than measuring this directly, it can be obtained from a model that updates its state solely from joint angle readings.

The physics-based model of the ECCERobot now developed (see Chapter 3) provides the best estimate currently available of the kinodynamic state of the complex robot structure under torque load. It is therefore a strong candidate to be incorporated into this controller, not only within the planner but as part of the noise reduction and delay compensation also. The fact that it is held within a standard physics-engine also provides a significant opportunity to integrate, in a single “scene”, the sampled environment state if it an appropriate sensing mechanism can be developed.

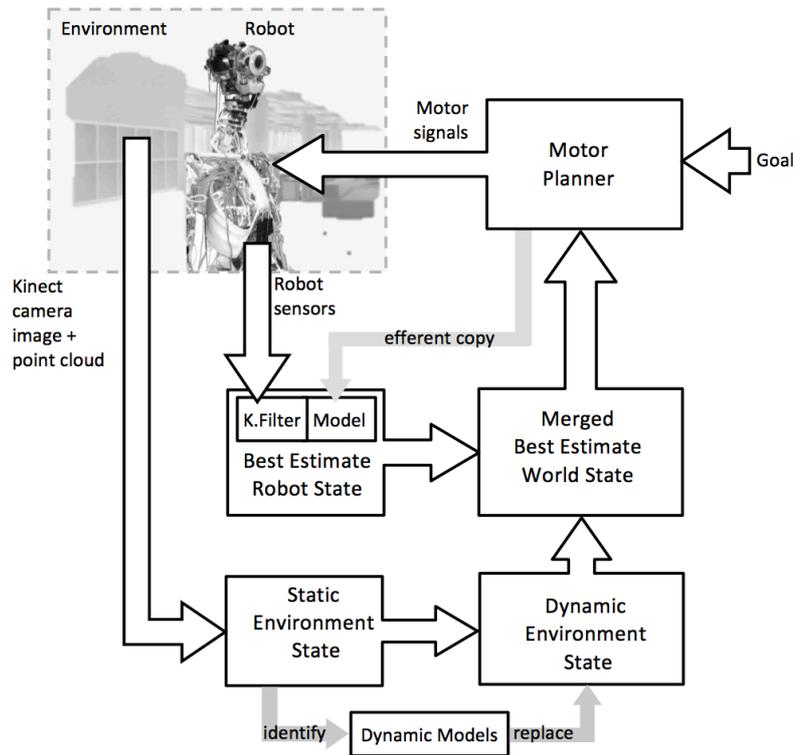
However, as it stands this model brings with it constraints of speed, accuracy and mathematical intractability due its non-linearity. These must be allowed for in the controller design, for example, to be able to use Kalman filtering for proprioceptive data correction would require a specialised non-linear approach (e.g. Wan & Van Der Merwe 2001) to function with such a model.

## 6.5 Proposed Design

Figure 42 shows a schematic of an initial controller design, based on the MPC principle comprising an iterative cycle of first capturing both the system under control and its environment into a forward model predictor, followed by a replanning based on the new best estimate state and the goal set for the system. Note that compensation for the delay in sensing, predicting and planning has not yet been incorporated in this design, however, it includes important modules for integrated proprioceptive and environment sensing, model-based prediction and motor planning. These are discussed now in turn.

### 6.5.1 Addressing sensor noise and inaccuracy

A standard approach to reduce the impact of inaccurate or noisy sensor readings is the use of Kalman filtering (Balakrishnan 1978; Welch & Bishop 2006). A forward model estimates the system state to obtain an alternative to the direct sensor sample. The



**Figure 42. MPC-based robot controller using a physics engine to capture and predict dynamic state of robot and environment**  
 Motor efferent copy and a model are used to generate a parallel estimate of the robots state. A Kalman filter is then used to find a best weighting between the estimate and the directly sensed but noisy sample state.  
 A Kinect sensor and object extraction techniques are added to capture a representation of the environment state with dynamics that can be merged with the physics-based robot model into a single unified “world” model. This is supplied to the planner enabling plans that allow for the environment, such as collision avoidance, to be generated.

Kalman filter (KF) algorithm is fed both the real and estimated signals and over time it will settle to an optimal balance, outputting the optimal estimated state.

In order for the model employed by the KF to estimate the new state it must have access to a copy of the control signals sent to the robot. Interestingly, in neurobiology, a copy of such motor efferent (“outwards”) nerves signals is observed (Sperry 1950; Kelso 1977), leading to speculation that such predictive models are also at work in the motor centres of the brain (Wolpert et al. 2001; Miall & Wolpert 1996).

Figure 42 shows that the best estimate robot state is obtained in our controller by combining sensor readings and motor efferent copy of the signals sent from the planner to the real robot. These signals are input a module comprising a Kalman filter and forward model to generate a best estimate robot state in the form of the physics-based model developed in Chapter 3.

### 6.5.1.1 *Kalman filtering with the ECCERobot physics-based model*

Although the KF principle is sound, we must however acknowledge that a serious challenge exists to implement this design. For individual robot sensors with a linear response a standard Kalman filter can be applied to each for noise reduction. However, for a robot structure where, for example, joint angles can have a non-linear dependency on other parts of the structure, it may be necessary to apply a single Kalman filter to the complete model alongside the full sampled state of sensor readings. Since this system is no longer linear the standard Kalman design cannot be employed and use of a non-linear Kalman filter is implied, such as the “unscented” Kalman filter (Wan & Van Der Merwe 2001). This takes the standard approach to handling non-linear systems by approximating it to an incremental set of linear systems. However, applying this approach to embed such a complex physics-model within a KF has not been successfully reported to date in the scientific literature and remains a goal for the future.

### 6.5.2 **Planning**

For a planning module we propose initially the learning controller developed (Chapter 4) for reaching control. This takes as input the current state of the robot plus the goal, as being the current location of the target object and, if desired, a designated hand. The planner generates as output the set of signals designed to take the designated hand to the target. However, to act as a continuous controller under this architecture we require an extension to the reaching controller allowing it to function from different starting states (see Chapter 4, section 4.3.5.3 *Extending the problem space – commencing from any state* ).

### 6.5.3 **Environment Capture**

For a generic continuous controller to succeed then it is critical to capture not only the robot’s own state, but sufficient of the (potentially rich and dynamic) environment state and also the relative position of one to the other. Indeed, for much of the history of robotics, it was arguably primarily this environment aspect that kept more complex robots in the sanitised (and flat-floored) laboratory or factory and away from the messy, “real” world.

In our controller design (Figure 42), the physics simulation “scene” or “world’ occupied by the robot model might also house a static, or even dynamic, model of the

immediate surroundings, if they could be effectively sampled and processed. This would result in a single integrated model comprising the robot situated within its modelled surroundings, thus allowing predictions to be generated that accounted for changes occurring outside the robot body and also those caused by interaction, such as objects affected by collision. This is therefore a potentially powerful approach, where even the classic “frame problem” (Korb 1998), might be reduced, if not nullified, by this “one world” design.

### **6.5.3.1 Use of Kinect Sensor and object extraction algorithms**

To begin to realize this design, a demonstration 3D vision system (Devereux et al. 2011) has been developed for the ECCERobot that employs a head-mounted Kinect sensor (Microsoft 2013) to capture unified depth map and colour photo data from the robot’s surroundings. The depth map is transformed into a static mesh and inserted as a set of collision shapes into the physics “world” alongside the robot model. In parallel, the colour image and depth map are processed with object extraction algorithms to recognize parts of the mesh that correspond to an object previously identified, for which a dynamic model has been constructed using the same physics engine. This section of the mesh is removed from the static model and replaced with this pre-designed dynamic physics model. The demonstration system performs this task by recognizing a real water bottle and replacing it in the static mesh within the physics engine with a model that the model robot can interact with.

## **6.6 Delays**

### **6.6.1 Effects of sensorimotor delay**

An important observation from the ECCERobot’s anthropomimetic predecessor Cronos, (Holland & Rob Knight 2006) was that its compliance and dynamic complexity made it very hard to employ open-loop control specifically because of the resultant unpredictability of its state. The body swayed and oscillated, and raising an arm or releasing a held object had far greater effect on the *kinodynamic* state (kinematics plus motion) than would be the case with a traditional stiff actuated robot. State captured via proprioceptive sensors often bore little kinodynamic resemblance to the state only a short while later, even without further control input. It was therefore expected that, for

the ECCERobot, even small delays between sensing and action would also cause a significant control issue.

### **6.6.2 Causes of delay**

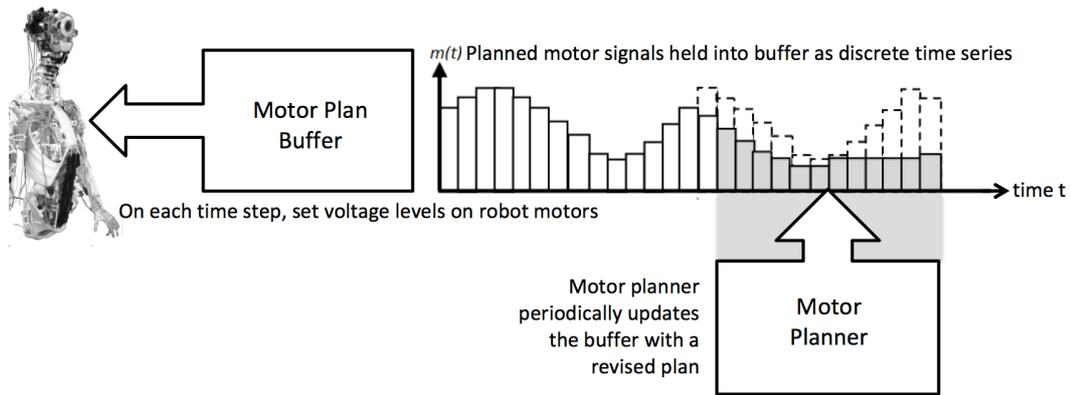
In any control system, the full delay between sensing and acting will build up across a number of stages. For example, consider the delay in capturing sensory data, transmitting the data for processing, updating an internal state representation and devising a motor plan based on this state and a goal. Finally there is the delay in transmitting the revised motor signals out to the motors or muscles.

### **6.6.3 Combating Delay**

Conventional system feedback control systems will often look to use high frequency, high precision sampling to directly minimise the sensorimotor delay and resultant instabilities (Franklin et al. 2002; Levine 1996). Alternatively, if the sensorimotor loop delay remains too large then predictive feedback approaches such as the Smith Predictor (Smith 1959; Franklin et al. 2002) may be employed, using a fast model to quickly estimate the feedback signal. However, in our case, the available physics-based model and planning are too slow for such a complex, dynamic control subject. We therefore turn again to strategies derived from Model Predictive Control. As discussed, MPC plans its control signals in extended iterations based on the predicted world state (robot plus environment) at a “horizon” point in the future. By allowing for the complete system sensorimotor delay when setting the horizon, we can compensate for slow modelling or planning elements in the loop (Valencia et al. 2011; Kobayashi & Hiraishi 2012). In theory, so long as these elements run faster than real-time then, so long as the state sampling and modelling remain perfect, then this approach will succeed. Of course, in practice this is not the case, therefore to best configure the system we must first look to characterise changes in the sensitivity of our proposed controller as the horizon point and modelling accuracy vary. This experiment is reported below (section 6.7).

### **6.6.4 Predicting intended motor signals with an updateable “buffer”**

For an MPC-based controller to function the motor efferent copy signal is not sufficient to predict system state up to the horizon point as this lies in the future. We therefore also need to be able to accurately predict the motor signals that will be output up to the



**Figure 43. Motor signal buffer design for an MPC-based controller for the ECCERobot**

Control signals are sent to the physical robot motors continuously, read from a single master buffer or queue that stores the current proposed motor a time series of motor signals. Each motor plan revision is loaded into the buffer, replacing the data from the point when the new plan can take effect, i.e. after the complete sensorimotor delay has passed.

horizon point. To address this, Figure 43 illustrates a proposed design for an updateable motor signal storage “buffer” that holds the current best motor plan as a time series of motor signals. The buffer continually passes the actual control signals to the physical robot whilst the planning system independently generates repeated motor replans that are written to the buffer at the place corresponding to the current horizon point.

The advantage of this parallel approach is that while sensors are read, predictions made and plans revised, the robot will simply continue to move under a known set of motor commands which comprised the best movement plan that could be generated at that time. The predictive forward model can therefore read ahead from the buffer to anticipate the motor signals into the future up to the chosen horizon point. This allows it to estimate the future world state once all delays have been accounted for.

### 6.6.5 Delay compensating design

Figure 44 shows the controller extended to incorporate the delay compensation mechanism. The important difference in this controller is that, as discussed, once the estimated state at the time  $t_{sense}$  of the sensor readings has been obtained, the motor signal buffer can be “read ahead” into the future in order to roll forward the integrated physics model of the world from the state  $S_{sense}$  (the output of the Kalman filter) to a predicted future state  $S'_{horizon}$  at time  $t_{horizon}$ .

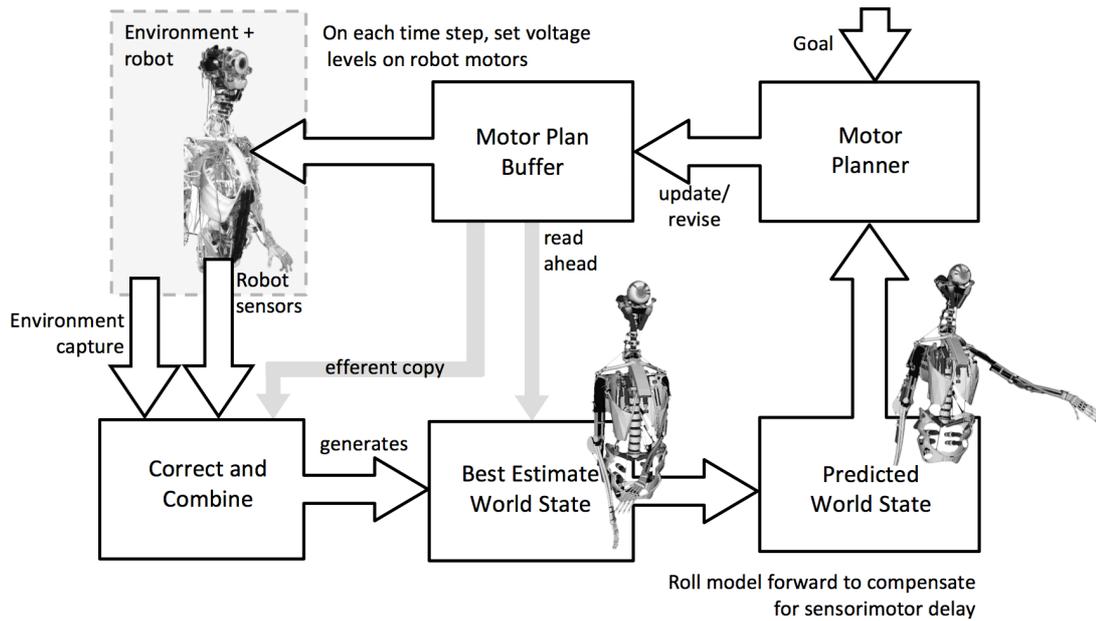


Figure 44. Schematic of full MPC-based controller for the ECCERobot incorporating delay compensation mechanism

We now plan a new time series of motor signals based on a starting state  $S'_{horizon}$  thus compensating for the change of world state that will have occurred during the period  $t_{horizon} - t_{sense}$ . However, this presents a problem. With a relatively slow and inaccurate model, we do not in practise know how precisely  $t_{horizon}$  must be set. Since sensorimotor delay may be difficult to measure accurately, or may vary between iterations, quantifying sensitivity of  $t_{horizon}$  to control success is a valuable exercise. In fact, some controller designs incorporate self-tuning compensation for this reason (Kobayashi & Hiraishi 2012).

Logic suggests that  $t_{horizon}$  should be set to the moment that a revised motor plan can begin to reach the actuators. However we have not proven this to be the case for an ECCERobot model-based controller, nor whether it is equally true for controllers with differing sensorimotor delay. For example, as we consume more time in prediction and planning, the future state  $S'_{horizon}$  will become correspondingly more inaccurate as modelling inadequacies are compounded. It may thus prove the case that a poorer plan, but computed faster, in fact performs better. We therefore conduct a simulation experiment to characterise the performance of the compensating controller of this robot under varying system delays.

Note also that this design neglects other potentially important factors that will affect accuracy of prediction. Firstly, in this design we do not include a mechanism, equivalent to the planning buffer, to predict the future of other, independent variables in the environment. Consider how a boxer might anticipate his *opponent's* moves from sensory clues and plan his own accordingly. Secondly, we also neglect, for now, the case where different sensors may have different delays and assume that these differences are not sufficiently significant to affect the state estimation.

## 6.7 Experiment exploring delay compensation

### 6.7.1 Overview

Using a second copy of the model robot as a surrogate for the real robot, we describe a relatively simple experiment to test the effect on control performance of the delay compensation elements of the design. In particular we seek to characterise the variation in performance with changes in the overall system delay, the compensation time and the simulation accuracy of the predictive model – the physics-based model of the ECCERobot in this case. Three principal findings emerge.

Firstly, the experiment clearly confirms that performance controlling this structure is maximised when delay compensation is matched precisely to the total system delay.

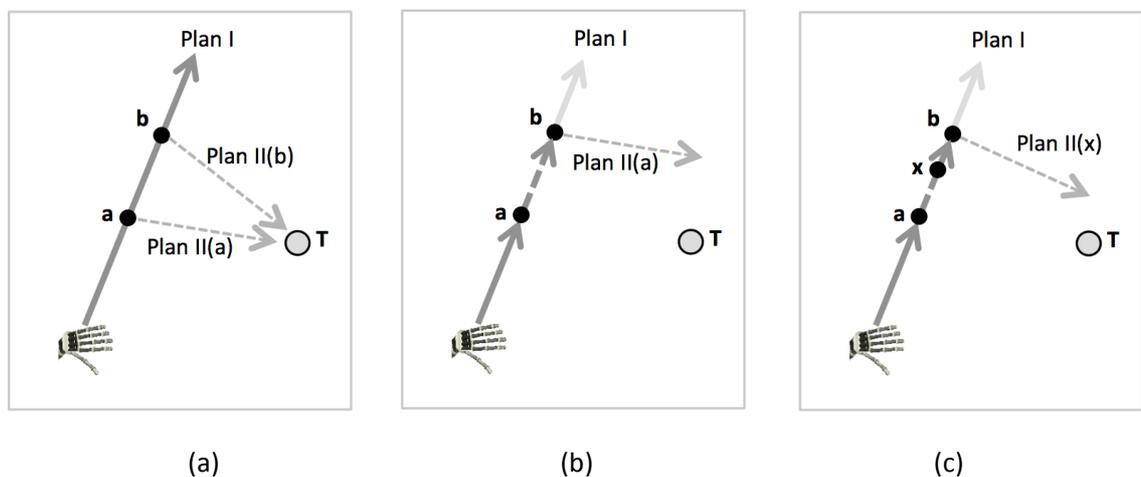
Secondly, if the compensation is mismatched to the actual delay, (too much or too little compensation), then performance degrades significantly (by approximately 40% for a 100ms mismatch).

Finally, although delay compensation has a significant effect, this is no more a factor than modelling inaccuracy itself, performance degrading by 30% for a 10% reduction in model accuracy (measured as the average parameter deviation imposed between model and surrogate robot). However, even on a 97.5% accurate model, if precise delay compensation is neglected, then the error is compounded and performance reduces steeply as uncompensated delay increases. Nevertheless, the problems arising from modelling accuracy call into question the use of a model-based controller for this robot unless the simulation time required by the prediction component of the delay compensation can be reduced to fraction of its current value.

### 6.7.2 Method

In order to induce planning errors caused by out-of-date state capture, we employ a scenario where the robot shifts from responding to a simple arm-raising motor plan to a second plan which moves a hand accurately to a target. By varying the planning delay, the amount of compensation applied and the model accuracy we can characterise the performance of the controller in managing this scenario under real world conditions where delays and model accuracy could significantly impact performance. Figure 46 schematically illustrates the scenario and shows the effects of using an uncompensated and a compensated plan.

To investigate this we employ two main steps. Firstly, we must train the controller to be capable of moving the hand to the target starting from any given position along its fixed trajectory generated by the initial arm raising plan. This training is performed under ideal conditions of zero delay, zero compensation and a perfect model of the surrogate robot.



**Figure 45. The impact of sensorimotor delay on planning and use of compensation**

The figure provides a conceptual illustration of an experiment based upon the control scenario where we wish the robot to shift from responding to a simple arm-raising motor plan (Plan I) to a second plan which moves a hand accurately to a target (Plan II).

(a) TRAINING. The planner is trained on an ideal zero delay system to move the hand to a fixed target T starting from any given position (such as a or b) along its fixed trajectory generated by the initial arm raising plan (Plan I). After training an effective plan can be generated by inputting the sensed hand position e.g:  $a \rightarrow \text{Plan II}(a)$ ,  $b \rightarrow \text{Plan II}(b)$ , etc.

(b) UNCOMPENSATED DELAY. The hand reaches position a, where it is sensed and passed to the planner, generating Plan I(a). During the resultant delay  $T_{\text{DELAY}}$  (sum of transmission, planning etc) however, the hand has continued under Plan I to reach position b. When the plan is finally applied the hand fails to reach T accurately.

(c) COMPENSATED DELAY. The hand reaches position a, where it is sensed. Before passing to the planner the model of the robot is clocked forward in time by a compensation period  $T_{\text{COMP}}$ . The predicted position X of the hand is passed to the planner, generating a Plan II(x) intended to be considerably closer to the ideal Plan II(b). By varying position a,  $T_{\text{DELAY}}$ ,  $T_{\text{COMP}}$  and the model accuracy we may characterise the behaviour of the controller under conditions of delay and compensation.

The second step tests the effects of altering the delay, compensation and model accuracy for each of 4 starting points selected along the arm-raising trajectory.

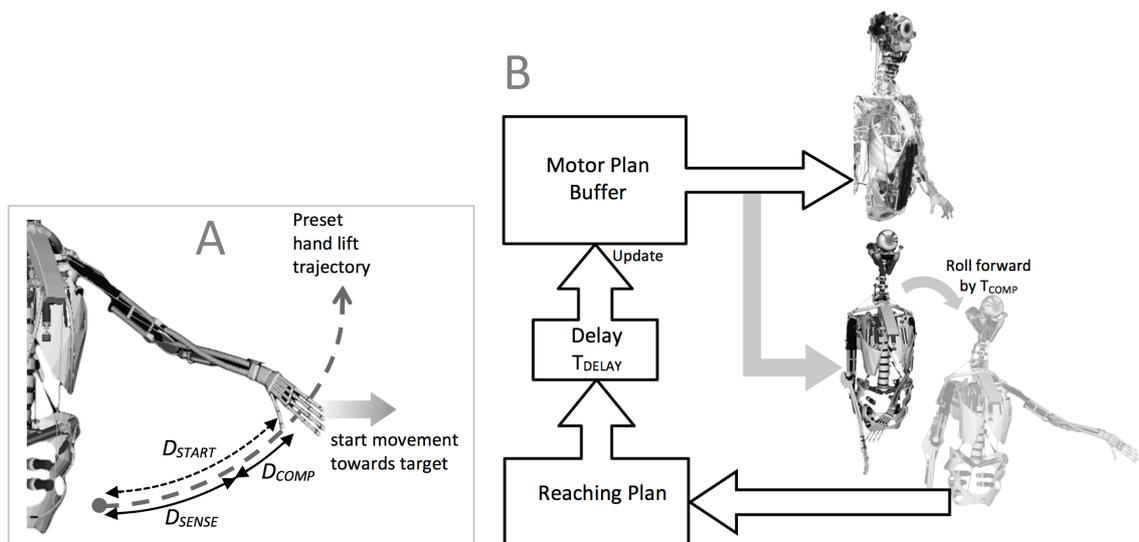
### 6.7.2.1 Step 1 – training under ideal conditions

The first step is therefore for the controller to learn to reach to the target from different starting hand positions. To do this, we employ again the synergy-based controller developed in Chapter 4. but we reverse the problem state; specifically we now use a *fixed target location* but a *varying start location* for the reaching hand.

All learning trials commence with the arm being raised under the same fixed motor plan  $M_{PRESET}$ , formed from a simple combination of synergies. Each trial commences by reaching towards the target from a different point along the preset hand trajectory generated by  $M_{PRESET}$ . The problem state can therefore be represented by the scalar distance  $D_{START}$  moved by the hand along this fixed trajectory until reaching movement to the target begins (see Figure 46A).

### 6.7.2.2 Step 2 – testing effects of delays, compensation and model accuracy

Once the controller has been trained to reach the fixed target reliably, starting from any position along the preset hand trajectory (see Step 1 above), we may commence testing the delay compensation design. The underlying strategy of the test is to simulate a set



**Figure 46. Experimental configuration to characterise the performance of the delay compensation design**  
 Figure A shows how a problem state is generated from a random generated along the preset hand lift.  
 Figure B shows how the controller generates a delayed reaching plan after the physics model has been rolled forward by a compensatory amount of time.

of four possible (total) system delay values that could exist in the sensorimotor loop, including the time to capture sensory data, transmit it, update the model state, combine with the environment state, predict future world state, plan a revised motor and update the motor signal buffer. We use  $T_{DELAY}$  to denote this notional total system delay.

To characterise the behaviour under compensation we plot, for each one of the four system delay settings, the success of the reaching behaviour over a range of compensation values. We are interested in whether the optimum compensation setting for this non-trivial control subject can be consistently inferred from the system delay and also in characterising the sensitivity of success to the precision with which delay is measured and the resultant compensation set.

The physics-based model and a second copy acting as a surrogate “robot” are reset to the same exact starting state and a random distance  $D_{SENSE}$  is generated, representing the distance that the hand is to be lifted before the state is sensed. In an *uncompensated* system this distance would be used directly as the problem state input to the reaching planner.

Figure 46b now illustrates the process. The known preset motor plan  $M_{PRESET}$  is loaded, as a time series of motor signals, into the buffer (see section 6.6.5) which outputs the motor control signals continuously to the “robot” and also to the model (i.e. acting as efference copy). Once the buffer begins this work, then after some period  $t_{SENSE}$  the hand will have travelled its required distance,  $D_{SENSE}$ . Note that the physics timestep remains at 3ms, the largest value where model stability could be maintained (see modelling Chapter 3, section 3.7).

At this point  $t_{SENSE}$  the simulations and the buffer feed are now paused and the “model” alone is clocked forwards by a delay compensation period  $T_{COMP}$  its behaviour determined by reading ahead, from the buffer, the upcoming motor signals. During this period, the model hand will move an additional distance,  $D_{COMP}$ . Note that we do not simply allow the buffer to drive both model and robot for this period  $T_{COMP}$  because we also wish to test the behaviour where  $T_{COMP} > T_{DELAY}$ , i.e. where we have over-compensated.

The total distance  $D_{START} = D_{SENSE} + D_{COMP}$  that the modelled hand has now travelled is now used as the driving problem state to generate a reaching plan, drawing upon on

the learning undertaken earlier. The new plan is now written into the buffer, but delayed by the system delay setting  $T_{DELAY}$ , i.e. by overwriting the current buffered plan starting from the time  $(t_{SENSE} + T_{DELAY})$ . It is critical to note that the buffer write always uses the system delay  $T_{DELAY}$ , not the compensation time,  $T_{COMP}$  which will be set to a different value on every trial.

With the robot simulation and the buffer feed now restarted, the robot hand continues on its preset lifting trajectory. The signals in the buffer forming the start of the actual reaching plan begin to take effect at time  $(t_{SENSE} + T_{DELAY})$ . The buffer continues, as the robot hand finally leaves its preset path to begin to reach for the target. Reward is accrued and the motor plan completes.

The simulations are then reset to the start conditions and the trial repeated 50 times against the same problem state with a new value of  $T_{COMP}$  which is incremented 10ms each time from 0ms to 200ms.

We test for each of four settings for the total system delay  $T_{DELAY}$  (0ms, 50ms, 100ms and 150ms). This scale was chosen to allow the furthest hand position that can be input to the planner (including the maximum compensation value of 200ms) to remain within the outer limit of the effective range. As we are interested in comparative performance, we do not plot the absolute reward but the fraction of the reward score that accrued when the same problem state was addressed using an ideal system without delay or compensation ( $T_{DELAY} = 0, T_{COMP} = 0$ ).

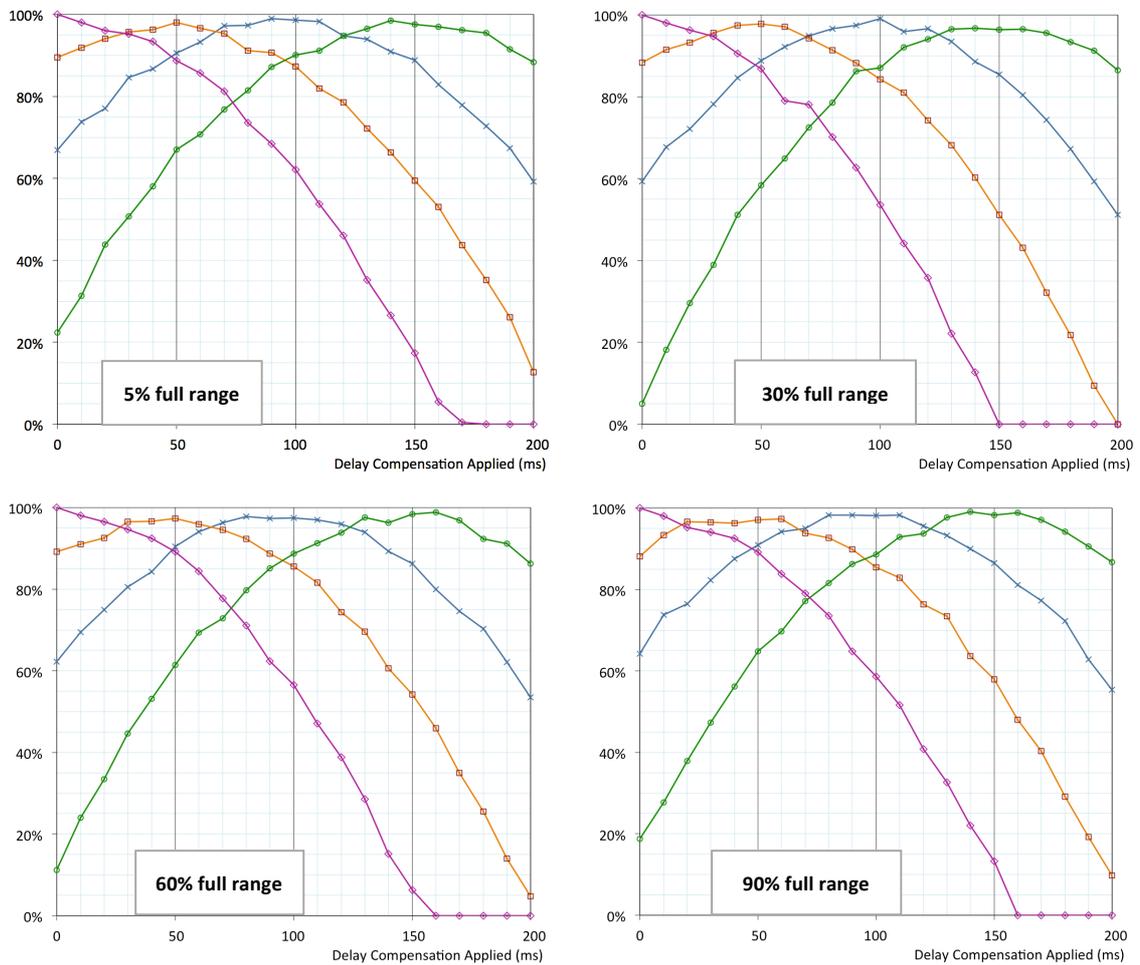
Note that as we are here interested in repeatable exploitation for comparisons, not exploration for learning, all random elements are removed from these tests, including generation of problem state and signal noise. We also test at four fixed problem states - corresponding to early, middle and late points in the preliminary hand-lifting stage ( $D_{SENSE} = 5\%, 30\%, 60\%, 90\%$ ) - in order to control for performance changes arising when reaching is commenced from different points.

### 6.7.2.3 Results

The results for the four chosen problem states ( $D_{SENSE} = 5\%, 30\%, 60\%, 90\%$ ) are shown in Figure 47. All four graphs consistently confirm that, as suggested (see section 6.6.5), the peak performance is obtained where the delay compensation matches the imposed system delay, i.e. where  $T_{DELAY} = T_{COMP}$ . When compared with an ideal (no-delay) controller, performance consistently falls off near-symmetrically for both over and

under-compensation in a steepening response curve, falling at approximately 30 percentage points per second in the first 50ms of mis-compensation, steepening to 150 percentage points lost per second after 100ms.

We also note that, for the accurate model the slightly flattened curve preceding the peak (e.g. see 100ms blue trace) shows that under-compensation performance holds up marginally better than over-compensation. We suggest this may be due to a greater



**Figure 47. Delay-compensated reaching performance over a range of fixed system delays and reaching commencement positions**

Each trace shows reaching performance variation over a range of delay compensation applied to a simulated robot system with a fixed system delay. Performance is measured relative to the reward obtained by a zero delay, zero compensation system reaching from the start position allocated to each figure. This is the first point plotted on the zero-delay (purple) trace.

Each figure shows four traces, each for a different setting for the fixed system delay set at 0ms (purple), 50ms (orange), 100ms (blue) and 150ms (green). The four figures repeat the same test, but use a different position along the trajectory of a lifting hand as the start point for a reaching movement to the target. The position allocated to each is indicated on the figure.

likelihood of striking the target nevertheless even if the hand should aim too high rather than too low. This will occur if the movement is planned using an earlier state (under-compensation), as the hand is relatively lower down in this case. Since the performance is measured in comparison with that obtained by an equivalent test of a zero delay system there is little change in the peak levels of the response graphs; however, the fall-off rate in performance varies to a small degree between the four charts, corresponding to different reaching start points. There is no clear pattern however, and may therefore be due to more or less performance sensitivity to the particular synergy weighting combination selected by the planner for the different start points.

### 6.7.3 Characterising Effects of Model Divergence

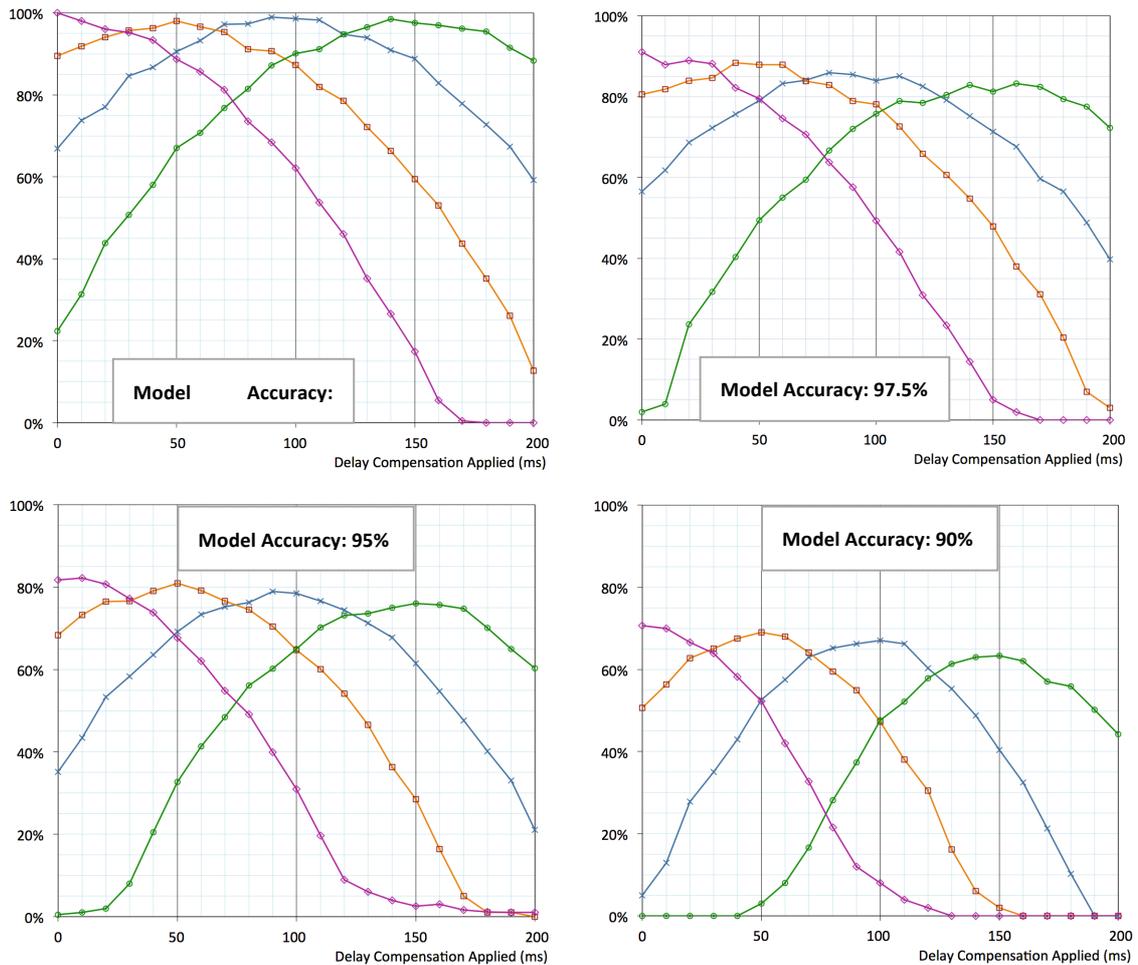
Up to this point, the experiment has employed a perfect match of model and surrogate “robot” which would not be possible using a real robot as no model would match it perfectly. We therefore also consider whether and how the relationship of  $T_{DELAY}$  and  $T_{COMP}$  changes with degrees of model divergence. To this end we generate three further robot surrogates by randomly varying (using a Gaussian distribution) a subset of model data by an average of 2.5%, 5% and 10% of their original values respectively, resulting in models of 97.5%, 95% and 90% accuracy. Note that we select model parameter values that can be altered without incurring lengthy modelling issues through morphology changes. We therefore include muscle attachment points, bone and motor weights, pulley locations and centre of gravity positions and maintain left/right mirroring whilst avoiding morphology parameters such as limb length.

For these altered models we employ only the earliest ( $D_{SENSE} = 5\%$  full range) of the four problem states that were tested for the perfect model. This is to minimise the divergence between the model and robot states before the problem state is even reached, recalling that the planner has been trained on the perfect model.

#### 6.7.3.1 Results

The results for performance under delay compensation for the three altered models are shown in Figure 48 alongside the unaltered model.

The first important observation is that the performance degradation due to model inaccuracy alone is surprisingly high for this structure, suggesting that, whilst delay compensation has a significant effect, modelling accuracy will be a very substantial



**Figure 48. Effects of model accuracy on delay-compensated reaching performance of simulated ECCERobot reaching**

Each plot shows reaching performance variation over a range of delay compensation applied to a simulated robot system with a fixed system delay. Performance is measured relative to the reward obtained by a zero delay, zero compensation system with a perfect (100%) model (the first point plotted on the purple trace, top left figure). Each figure shows four plots, each for a different setting for the fixed system delay set at 0ms (purple), 50ms (orange), 100ms (blue) and 150ms (green). The four figures repeat the same test, using the same hand position for the commencement of reaching to the target, but for each a different model is used for planning, matching the surrogate robot model with an accuracy indicated on the figures. Model variation is obtained by randomly varying a set of the model parameters using a Gaussian distribution with a mean equating to a fixed fraction above or below the robot model's equivalent parameter value.

factor in the success of this controller design. For example, an optimally compensated system with model accuracy of 90% performs at around 71% of a perfect model, equivalent to the effect of a 104ms under-compensation for a perfect model. However the compound effect of both a lack of delay compensation and slightly inaccurate modelling shows a far larger drop to only 5% of a zero-delay, perfect model. This suggests that delay-compensation must be considered a particularly important strategy if models are not highly accurate, which is almost certain to be the case with a real robot.

Once again, as expected, the peak performance occurs where  $T_{DELAY} = T_{COMP}$ . However, we now see consistently steeper curves as model accuracy decreases, in particular on the over-compensation side, showing that sensitivity to the compensation setting is increasing. This can be explained by considering that the model state is also diverging from the robot's during the excess compensation period.

For each of the inaccurate models we also now note a reduction in the peak performance as higher settings for system delay are employed, totalling on average a 5.8% drop from zero delay through to 150ms. This is not surprising if we again consider the increasing amount that the model state is being rolled forward in time before the reaching movement is planned. This means the real and modelled states have diverged further causing an inaccurate reaching plan to be generated.

## 6.8 Discussion and Conclusion

We have derived and presented a design for a continuous controller for the ECCERobot. The design was selected for compatibility with the model already developed for the robot motor planner and is grounded on the proven approach of model predictive control but contains novel elements, particularly in the context of controlling of such a complex, musculoskeletal robot. These include a physics-based forward model, a Kinect sensor-based vision system for 3D mesh capture of the robot's surroundings and identification of dynamic elements, integration of the environment within the same physics engine and a muscle synergy based motor planner.

Although elements of the design remain unproven to date, particularly the integration of Kalman filtering with a complex physics-based forward model, a pilot version of the environment capture system has been developed (Devereux et al. 2011) and the effects of modelling accuracy and delay compensation design are characterised in experiments presented here.

These experiments have demonstrated a significant benefit of applying delay-compensation techniques to a model-based controller for a structure such as the ECCERobot. The results support the principle that a precise match between compensation and overall system delay provides the best performance and that sensitivity is high for this complex simulated structure, the performance degrading rapidly with over or under compensation.

However, for such a structure, the performance degradation suffered by inaccurate modelling is also considerable, such that doubt must be cast upon the benefit of a forward model-based controller approach in cases where the prediction and modelling overheads themselves contribute significantly to the overall system delay. For a physics-engine based model this is further compounded since the prediction delay is not fixed but itself increases with the amount of time requiring simulation.

In the case of the relatively inaccurate and slow performing model of the ECCERobot developed to date (running at approximately real time, see Chapter 3 - Modelling) a strong case can be made for directing short term efforts towards reinforcement learning based on the robot itself (see Chapter 4 – Future Work) and eliminating the physics model from the controller. This approach lacks explicit delay compensation, yet may allow indirectly for its effects by mapping reward generated directly to the sensed starting state, regardless of how much later the motor plan actually activates. However, for any problem where the starting state can be dynamic this would likely require an extension of the state capture, adding some kinodynamic elements such as the hand velocity vector (see also section 5.7.3).

Nevertheless, in the longer term, when a significantly faster simulation becomes available- most likely via GPU acceleration - then the benefits of passing an integrated simulated world state (robot plus environment) to the planner will likely prove the most rewarding path.

We also suggest that where a robot under control has dynamics sufficiently simple to allow fast physics-based modelling and Kalman filtering then this architecture - harnessing three dimensional environment capture, a physics engine with a merged model of the robot and environment, plus a delay compensation mechanism - holds considerable promise as a generic control solution.

Finally, it is intriguing to consider potential parallels with human perception in motor control tasks. As we have discussed, the brain appears far better at compensating for delay issues than any comparable artificial controller, and we have shown that a successful control mechanism for a complex compliant structure with high sensitivity to sensorimotor delays can be one that drives its planning from a predicted state. A range of studies in both cognitive science and neurobiology directly support the notion that the self-perceived “current state” employed for planning a motor task may not

comprise the state captured at the moment of sensory input but rather a prediction. For example, subjects performing a motor movement were found to be more conscious of the relevant point in their planned movement than their actual movement – which they had been induced to unconsciously distort (Fournieret & Jeannerod 1998).

In general, supporters of this theory previously have postulated that incoming sensory information represents an out-dated state from the past, therefore a predictive forward model driven from motor efferent copy can be used to estimate the state “now” which is then used for the basis of action selection.

However, what is interesting is that our controller appeared to have no need for the state of the robot “now”. Instead, a past state is rolled directly forward to a *future* state for use in planning, in order to compensate for all delays. It is therefore intriguing to speculate that what we ourselves consciously perceive as “how things are right now” may, in fact, comprise a prediction of the world in the near future.

## **Chapter 7 :**

### **Conclusion**

---

#### **7.1 Aims of the thesis**

The aim of this thesis was to begin to develop and test an effective control approach for anthropomorphic robots, such as the ECCERobot, a musculoskeletal, biomimetic humanoid torso. This new class of robots have the potential to be deployed far more safely in human environments than their conventional stiff counterparts but offer a significant challenge to conventional control approaches.

We therefore examined the particular issues of the associated control problem and considered a number of established or emerging control approaches, including evidence from biological motor systems. We conclude that bio-inspired approaches hold the most promise for controlling a biomimetic structure that would be considered highly challenging by conventional robot controllers.

We consequently reviewed in greater detail a range of bio-inspired approaches with a view to selecting for investigation one with a strong combination of novelty, promise, and interest. In particular, we focused upon recent strong evidence from biological studies demonstrating the extent to which effective motor control of frogs, cats or humans appears to draw heavily upon a combination of advantageous, co-evolved natural dynamics and simple fixed-weight activations of precise muscle groupings (synergies).

We concluded from the evidence that a promising and relatively novel study would test the hypothesis that drawing upon a muscle group co-activation approach for an extensive biomimetic robot structure with potentially rich natural dynamics may facilitate significantly simpler search and learning techniques to be deployed than the complex algorithms currently under development for generic, high-dimensional control subjects.

Of these simpler methods, we chose to trial an approach built primarily from reinforcement learning (RL) fundamentals, citing as reasons its bio-inspired nature and “action discovery” potential for exploiting natural dynamics of the full body. We therefore proposed, since effective synergy patterns for a given musculoskeletal robot would be unknown, to derive a simple reinforcement learning approach intended to allow these patterns to emerge, in particular those that aid linearization of the control. We also sought to draw upon optimal control theories to encourage the emergence of smoother, more natural movement by incorporating signal dependent noise and trial repetition.

We also considered whether our selected approach should be developed against the physical robot or a modelled approximation, at least for preliminary investigations. We briefly reviewed available full body models and musculoskeletal model building tools, concluding that none were fit for the purpose of an anthropomorphic robot controller. We therefore proposed employing a fast, modern physics simulation engine to construct a complete physics-based model which incorporates actuation modelling, demonstrates full body natural dynamics and can potentially predict dynamic interaction (e.g. collision) with sensed environment objects.

Finally, we considered the problem of designing a continuous control architecture for this class of robots and whether the physics model developed could be reused, not only as a motor planner, but to assist with significant standard control issues such as corrective state estimation and delay compensation.

## **7.2 Original Contributions of Thesis**

Here we summarise and defend the original contributions asserted in this thesis and associated published work.

### **7.2.1 A physics-based forward model of a complete musculoskeletal robot torso**

Although numerous biomechanical models of individual body parts or regions exist, both as simplified/idealised forms and detailed biologically-based musculoskeletal simulations, very few full-body human models exist and none of comparative musculoskeletal robots. Those tools that exist (e.g. *AnyBody*<sup>TM</sup>) are not designed as control platforms, but as medical or sporting tools and take as input real captured motions rather than muscle activation signals. For the ECCERobot we required a open

source based and relatively fast simulation model of a complex hand-built robot which, by necessity, is constructed with real materials and constraints as an engineering approximation to a human.

We have therefore developed an open-source, physics-engine based model of a complete musculoskeletal robot torso, reverse-engineered from the anthropomorphic ECCERobot, which was constructed using Grays Anatomy as a guide. The model is based on the standard *Bullet Physics* engine (Coumans n.d.) and adds a number of custom-modelled components include the elastic muscles, motors, gearboxes, pulleys and joint friction. A stable model is available with 55 elastic muscles and 88 degrees of freedom that can act as a biomimetic structure of high complexity. The model is implemented in standard C++ and runs in real time on a standard, albeit high-end, Linux PC (see Diamond & Holland 2012; Wittmeier et al. 2012; Wittmeier et al. 2011).

### **7.2.2 Simple reinforcement learning can produce reaching control of complex, musculoskeletal robot model by using an approach of muscle co-activations, simple shared driving signals and natural dynamics**

There are few studies published to date of synergy-based controllers leveraging natural dynamics in biomimetic musculoskeletal structures. We surmise this is because both the controlled subject (in robotics, at least) and the synergy approach remain relatively unconventional for now and because the biological data supporting the widespread existence of this simple control approach in nature is relatively new and conclusions remains disputed (Tresch & Jarc 2009; Kutch et al. 2008; Valero-Cuevas et al. 2009; Ting & McKay 2007). Few studies consider synergies across diverse body parts, such the arm and torso muscles (e.g. Ma & Feldman 1995). Of all the studies considered, none address an extensive humanoid body model with associated body-wide dynamics, focusing almost exclusively on body part models; of primarily the frog leg (e.g. Berniker et al. 2009), the human arm (e.g. Fagg et al. 2002), or the human leg (e.g. Neptune et al. 2009). Only one study, focused on modelling the cerebellum, combines synergies with reinforcement learning (Fagg et al. 2002) and then only to locate combinations of pre-rolled generic synergy patterns to control a very simplified model of an arm. No studies located apply this approach to musculoskeletal robots.

We therefore suggest that we have devised a promising, relatively simple, but effective control approach for a complex, full-torso, musculoskeletal, biomimetic humanoid structures by employing a novel combination of bio-inspired approaches, namely:

weighted muscle co-activation patterns, simple shared driving signals, reinforcement learning and natural dynamics. The approach has been shown to be effective in controlling a complex physics modelled simulation of a complete anthropomorphic robot to produce reaching to sequentially presented, randomly positioned targets.

### **7.2.3 A low dimensional reaching controller for biomimetic musculoskeletal modelled robot based on extracted emergent synergies**

We have also demonstrated that a set of emergent set of implied “candidate synergy” fragmentary patterns can be extracted from the learned full motor-co-activation plans and that these may be re-used directly in the same learning controller, achieving lower dimensionality by replacing individually activated muscles with these synergy patterns. This was found to both speed learning and performance level of the same task and to extend capability relatively rapidly to other reaching-related tasks requiring control of different dynamic forms triggered in the structure. By characterising the behaviour of this synergy-based controller when learning a range of tasks we conclude that it may be applied to rapidly assimilate other tasks, but that the level of performance drops commensurately with divergence from the original simple task.

This form of muscle-based control based on *extracted* synergies has been primarily studied with modelled frog legs. The stand-out comparable study of this kind (Berniker et al. 2009) directly analyses the natural dynamics of a biomimetically-modelled frog leg resulting in a low dimensional model. Key synergies that best control these dimensions with a linear response are identified and employed in an effective low dimensional controller. We suggest these results underpin and support our own findings but without duplication as they were obtained via an alternative approach and employed a very different, but still bio-inspired, control subject.

### **7.2.4 Optimal control principles can be exploited through RL trial repetition to refine movements**

We also offer some experimental evidence supporting the idea that the issuing of RL reward across repeated trials against the same problem state can bring about increased endpoint reliability of a reaching movement under signal-dependent Gaussian motor noise, resulting in more naturalistic movement of a biomimetic structure - as judged by chi-squared similarity to the well known bell-curve velocity profile observed in nature.

### **7.2.5 Biological implications: support for synergy-based motor control theories**

We conclude by arguing that the work offers some implications in understanding motor learning in biology. This primarily comprises strong support for theories espousing the effectiveness of synergy-based muscle control over highly complex, compliant, musculoskeletal structures. We have also demonstrated that the emergence of such synergies under relatively simple reinforcement learning, a form of learning also strongly implicated in the brain (Schultz 1998; Schultz 2002; Chorley & Seth 2011).

### **7.2.6 An MPC-based design for continuous control of an anthropomorphic robot incorporating delay compensation**

We have derived a design for a continuous controller for the anthropomorphic ECCERobot, incorporating the model already developed for the robot motor planner. The design is grounded on the proven approach of model predictive control but contains novel elements, particularly in the context of controlling of such a complex, musculoskeletal robot. These include a physics-based forward model, a Kinect sensor-based vision system for 3D mesh capture of the robot's surroundings and identification of dynamic elements, integration of the environment within the same physics engine and a muscle synergy based motor planner. Whilst elements of the design remain unproven to date, particularly the integration of Kalman filtering with a complex physics-based forward model, a pilot version of the environment capture system has been developed (Devereux et al. 2011) and the effects of modelling accuracy and delay compensation design are characterised in an experiment demonstrating a significant benefit of applying delay-compensation techniques to a model-based controller for a structure such as the ECCERobot. We find that a precise match between compensation and overall system delay provides the best performance but that sensitivity is high for this complex simulated structure, the performance degrading rapidly with over or under compensation.

We also find that with such a highly non-linear model performance degradation suffered by inaccurate modelling is also considerable, such that doubt must cast upon the benefit of a forward model-based controller approach in cases where the prediction and modelling overheads themselves contribute significantly to the overall system delay.

Finally, we suggest that where the robot under control has dynamics sufficiently simple to allow fast physics-based modelling and Kalman filtering then the

architecture design proposed here - harnessing a physics model, environment capture and delay compensation - holds considerable promise as a generic control solution.

### **7.3 Future Work**

Recommendations for future work to refine the reaching control algorithm and apply it to the physical robot are detailed in Chapter 5 (section 5.7). Recommendations for the MPC-based continuous controller are provided in Chapter 6 (section 6.8).

## Chapter 8 :

### References

---

- Ariff, G. et al., 2002. A real-time state predictor in motor control: study of saccadic eye movements during unseen reaching movements. *Journal of Neuroscience*, 22(17), pp.7721–7729.
- Atherton, D., 2006. Basic Nonlinear Control Systems. In E. D. Sontag & M. Thoma, eds. *Simulation*. EOLSS, pp. 547–562.
- Balakrishnan, A. V., 1978. The Kalman Filter. In J. Benesty, M. M. Sondhi, & Y. Huang, eds. *The Mathematical Intelligencer*, 1(2), pp.90–92.
- Barto, A., 1995. Reinforcement learning: Reinforcement learning in motor control. In M. A. Arbib, ed. *The handbook of brain theory and neural networks*. MIT Press, pp. 804–813.
- Beer, R., 1995. On the Dynamics of Small Continuous-Time Recurrent Neural Networks. *Adaptive Behavior*, 3(4), pp.469–509.
- Bellman, R., 1954. Some Problems in the Theory of Dynamic Programming. *Econometrica*, 22(1), pp.37–48.
- Berniker, M. et al., 2009. Simplified and effective motor control based on muscle synergies to exploit musculoskeletal dynamics. *Proceedings of the National Academy of Sciences of the United States of America*, 106(18), pp.7601–7606.
- Bienenstock, E., Cooper, L. & Munro, P., 1982. Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *Journal of neuroscience*, 2(1), pp.32–48.
- Bizzi, E. et al., 2008. Combining modules for movement. *Brain research reviews*, 57(1), pp.125–33.
- Blakemore, S., Wolpert, D. & Frith, C., 2000. Why can't you tickle yourself? *NeuroReport*, 11(11), pp.R11–R16.

- Te Boekhorst, R., Lungarella, M. & Pfeifer, R., 2003. Dimensionality reduction through sensory-motor coordination. *Artificial Neural Networks and Neural Information Processing – Lecture Notes in Computer Science*, 2714, pp.496–503.
- Bongard, J., 2009. Biologically Inspired Computing. *IEEE Computer*, 42(4), pp.95–98.
- Bongard, J., 2011. Morphological change in machines accelerates the evolution of robust behavior. *Proceedings of the National Academy of Sciences of the United States of America*, 108(4), pp.1234–1239.
- Brashers-Krug, T., Shadmehr, R. & Bizzi, E., 1996. Consolidation in human motor memory. *Nature*, 382(6588), pp.252–255.
- Brooks, R., 1991a. Intelligence Without Reason. In *Artificial intelligence critical concepts*. Taylor & Francis, pp. 569–595.
- Brooks, R., 1991b. Intelligence without representation. *Artificial Intelligence*, 47(1-3), pp.139–159.
- Burns, B. & Brock, O., 2007. Single-Query Motion Planning with Utility-Guided Random Trees. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, pp. 3307–3312.
- Carrillo, R. et al., 2008. A real-time spiking cerebellum model for learning robot control. *Bio Systems*, 94(1-2), pp.18–27.
- Cheung, V. et al., 2005. Central and sensory contributions to the activation and organization of muscle synergies during natural motor behaviors. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 25(27), pp.6419–34.
- Cheung, V. et al., 2009. Stability of muscle synergies for voluntary actions after cortical stroke in humans. *Proceedings of the National Academy of Sciences of the United States of America*, 106(46), pp.19563–19568.
- Choi, Y., You, B. & Oh, S., 2004. On the stability of indirect ZMP controller for biped robot systems. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 1966–1971.
- Chorley, P. & Seth, A., 2011. Dopamine-signaled reward predictions generated by competitive excitation and inhibition in a spiking neural network model. *Frontiers in computational neuroscience*, 5(May), p.21.
- Choset, H. et al., 2005. Principles of Robot Motion: Theory, Algorithms, and Implementations [Book Review]. *IEEE Robotics & Automation Magazine*, 12(3), pp.110–110.

- Cliff, D., Harvey, I. & Husbands, P., 1993. Incremental evolution of neural network architectures for adaptive behaviour. In *Behaviour*. Citeseer, pp. 39–44.
- Collewijn, H., Erkelens, C. & Steinman, R., 1988. Binocular co-ordination of human horizontal saccadic eye movements. *The Journal of Physiology*, 404(1), pp.157–182.
- Coumans, E., Bullet Physics Engine. Available at: [www.bulletphysics.org](http://www.bulletphysics.org).
- Cueli, J. & Bordons, C., 2008. Iterative nonlinear model predictive control. Stability, robustness and applications. *Control Engineering Practice*, 16(9), pp.1023–1034.
- D'Avella, A. et al., 2006. Control of fast-reaching movements by muscle synergy combinations. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 26(30), pp.7791–810.
- D'Avella, A. et al., 2008. Modulation of phasic and tonic muscle synergies with reaching direction and speed. *Journal of neurophysiology*, 100(3), pp.1433–54.
- D'Avella, A. & Bizzi, E., 2005a. Shared and specific muscle synergies in natural motor behaviors. *Proceedings of the National Academy of Sciences of the United States of America*, 102(8), pp.3076–3081.
- D'Avella, A. & Bizzi, E., 2005b. Shared and specific muscle synergies in natural motor behaviors. *Proceedings of the National Academy of Sciences of the United States of America*, 102(8), pp.3076–81.
- D'Avella, A., Portone, A. & Lacquaniti, F., 2011. Superposition and modulation of muscle synergies for reaching in response to a change in target location. *Journal of neurophysiology*, 106(6), pp.2796–812.
- D'Avella, A., Saltiel, P. & Bizzi, E., 2003. Combinations of muscle synergies in the construction of a natural motor behavior. *Nature neuroscience*, 6(3), pp.300–8.
- D'Avella, A. & Tresch, M., 2002. Modularity in the motor system: decomposition of muscle patterns as combinations of time-varying synergies T. G. Dietterich, S. Becker, & Z. Ghahramani, eds. *Advances in Neural Information Processing Systems 14*, 14, pp.141–148.
- Demiris, Y. & Meltzoff, A., 2008. The Robot in the Crib: A Developmental Analysis of Imitation Skills in Infants and Robots. *Infant and child development*, 17(1), pp.43–53.
- Der, R., 1999. Emergent robot behavior from the principle of homeokinesis. In *Experiments with the MiniRobot Khepera Proceedings of the 1st International Khepera Workshop99*. Citeseer.

- Desmurget, M. & Grafton, S., 2000. Forward modeling allows feedback control for fast reaching movements. *Trends in Cognitive Sciences*, 4(11), pp.423–431.
- Devereux, D. et al., 2011. Using the Microsoft Kinect to model the environment of an anthropomimetic robot. *Proc. of the 2nd IASTED Intl. Conf. on Robotics (Robo2011)*, Pittsburgh, USA.
- Diamond, A. et al., 2012. Anthropomimetic Robots: Concept, Construction and Modelling. *International Journal of Advanced Robotic Systems*. ISBN: 1729-8806, InTech, DOI: 10.5772/52421.
- Diamond, A., Holland, O. & Marques, H., 2011. The role of the predicted present in artificial and natural cognitive systems. *Frontiers in Artificial Intelligence and Applications*, 233: Biolo, pp.88–95.
- Drew, T., Kalaska, J. & Krouchev, N., 2008. Muscle synergies during locomotion in the cat: a model for motor cortex control. *The Journal of Physiology*, 586(Pt 5), pp.1239–1245.
- Eagleman, D. & Sejnowski, T., 2007. Motion signals bias localization judgments : A unified and Frohlich illusions. *Neurobiology*, 7, pp.1–12.
- Erbatur, K. & Kurt, O., 2009. Natural ZMP Trajectories for Biped Robot Reference Generation. *IEEE Transactions on Industrial Electronics*, 56(3), pp.835–845.
- Fagg, A. et al., 2002. A model of cerebellar learning for control of arm movements using muscle synergies. In *1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97*. IEEE Comput. Soc. Press, pp. 6–12.
- Feldman, A. et al., 1998. Tests of the Equilibrium Point Hypothesis. *Motor Control*, 2(3), pp.189–205.
- Flanagan, J. et al., 1999. Composition and decomposition of internal models in motor learning under altered kinematic and dynamic environments. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 19(20), p.RC34.
- Flanagan, J. & Wing, A., 1997. The role of internal models in motion planning and control: evidence from grip force adjustments during movements of hand-held loads. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 17(4), pp.1519–28.
- Flash, T. & Hochner, B., 2005. Motor primitives in vertebrates and invertebrates. *Current opinion in neurobiology*, 15(6), pp.660–6.

- Fourneret, P. & Jeannerod, M., 1998. Limited conscious monitoring of motor performance in normal subjects. *Neuropsychologia*, 36(11), pp.1133–40.
- Franklin, G., Powell, J. & Emami-Naeini, A., 2002. *Feedback Control of Dynamic Systems*, Addison Wesley.
- Ganor, I. & Golani, I., 1980. Coordination and integration in the hindleg step cycle of the rat: kinematic synergies. *Brain research*, 195(1), pp.57–67.
- García, C., Prett, D. & Morari, M., 1989. Model predictive control: Theory and practice—A survey. *Automatica*, 25(3), pp.335–348.
- Giszter, S., Mussa-Ivaldi, F. & Bizzi, E., 1993. Convergent force fields organized in the frog's spinal cord. *Journal of Neuroscience*, 13(2), pp.467–91.
- Gomi, H. & Kawato, M., 1996. Equilibrium-point control hypothesis examined by measured arm stiffness during multijoint movement. *Science*, 272(5258), pp.117–20.
- Gottlieb, G., 1998. Rejecting the equilibrium-point hypothesis. *Motor control*, 2(1), pp.10–2.
- Gravato Marques, H. et al., 2013. Self-organization of reflexive behavior from spontaneous motor activity. *Biological cybernetics*, 107(1), pp.25–37.
- Gray, H., 1901. *Grays Anatomy (Classic Collector's Edition)* P. T. Pick & R. Howden, eds., Bounty Books.
- Grossberg, S., 1995. The Attentive Brain. *American Scientist*, 83, pp.438–449.
- Grossberg, S. & Myers, C., 2000. The resonant dynamics of speech perception: interword integration and duration-dependent backward effects. *Psychological review*, 107(4), pp.735–67.
- Grush, R., 2004. The emulation theory of representation: Motor control, imagery, and perception. *Behavioral and Brain Sciences*, 27(03), pp.377–396.
- Gu, X. & Ballard, D., 2006a. Motor Synergies for Coordinated Movements in Humanoids. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 3462–3467.
- Gu, X. & Ballard, D., 2006b. Robot Movement Planning and Control Based on Equilibrium Point Hypothesis. In *2006 IEEE Conference on Robotics, Automation and Mechatronics*. IEEE, pp. 1–6.

- Hamilton, A., Jones, K. & Wolpert, D., 2004. The scaling of motor noise with muscle strength and motor unit number in humans. *Experimental Brain Research*, 157(4), pp.417–430.
- Harris, C. & Wolpert, D., 1998a. Signal-dependent noise determines motor planning. *Nature*, 394(6695), pp.780–4.
- Harris, C. & Wolpert, D., 1998b. Signal-dependent noise determines motor planning. *Nature*, 394(6695), pp.780–4.
- Hart, C. & Giszter, S., 2010. A neural basis for motor primitives in the spinal cord. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 30(4), pp.1322–36.
- Hart, C. & Giszter, S., 2004. Modular premotor drives and unit bursts as primitives for frog motor behaviors. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 24(22), pp.5269–82.
- Haruno, M., Wolpert, D. & Kawato, M., 2001. Mosaic model for sensorimotor learning and control. *Neural Computation*, 13(10), pp.2201–2220.
- Hebb, D., 1950. *The Organization of Behavior; A Neuropsychological Theory* Erlbaum, ed. *The American Journal of Psychology*, 63(4), p.633.
- Hinder, M.R. & Milner, T.E., 2003. The case for an internal dynamics model versus equilibrium point control in human movement. *The Journal of Physiology*, 549(Pt 3), pp.953–963.
- Hitomi, K. et al., 2006. Reinforcement learning for quasi-passive dynamic walking of an unstable biped robot. *Robotics and Autonomous Systems*, 54(12), pp.982–988.
- Holland, O. & Knight, R., 2006. The Anthropomorphic Principle. In *Proceedings of the AISB06 Symposium on Biologically Inspired Robotics*.
- Von Holst, E. & Mittelstaedt, H., 1950. Das Reafferenzprinzip. *Naturwissenschaften*, 37(20), pp.464–476.
- Ingram, J. et al., 2008. The statistics of natural hand movements. *Experimental Brain Research*, 188(2), pp.223–36.
- Ivanenko, Y. et al., 2005. Coordination of locomotion with voluntary movements in humans. *Journal of Neuroscience*, 25(31), pp.7238–7253.
- Izhikevich, E., 2007. Solving the distal reward problem through linkage of STDP and dopamine signaling. *Cerebral Cortex*, 17(10), pp.2443–52.

- Izhikevich, E. & Desai, N., 2002. The Relationship Between Spike-Timing-Dependent Plasticity (STDP) and Sliding Threshold (BCM) Synaptic Modification. *Neurosciences*, pp.1–2.
- Jäntschi, M., Wittmeier, S. & Knoll, A., 2010. Distributed control for an anthropomorphic robot. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 5466–5471.
- Jones, K., Hamilton, A. & Wolpert, D., 2002. Sources of signal-dependent noise during isometric force production. *Journal of Neurophysiology*, 88(3), pp.1533–1544.
- Kagami, S. et al., 2003. Humanoid Arm Motion Planning using Stereo Vision and RRT Search. In *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*. pp. 2167–2172.
- Kargo, W. et al., 2010. A Simple Experimentally Based Model Using Proprioceptive Regulation of Motor Primitives Captures Adjusted Trajectory Formation in Spinal Frogs. *Journal of Neurophysiology*, 103(1), pp.573–590.
- Kargo, W. & Giszter, S., 2000. Rapid correction of aimed movements by summation of force-field primitives. *Journal of Neuroscience*, 20(1), pp.409–26.
- Kavraki, L. et al., 1996. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4), pp.566–580.
- Kawato, M., 1999. Internal models for motor control and trajectory planning. *Current opinion in neurobiology*, 9(6), pp.718–27.
- Kawato, M. & Gomi, H., 1991. Model of four regions of the cerebellum. In *Proceedings of the 1991 IEEE International Joint Conference on Neural Networks*. IEEE, pp. 410–419.
- Kelso, J., 1977. Planning and efferent components in the coding of movement. *Journal of Motor Behavior*, 9(1), pp.33–47.
- Kilgard, M. & Merzenich, M., 1995. Anticipated stimuli across skin. *Nature*, 373(6516), p.663.
- Klein-Breteler, M., Meulenbroek, R. & Gielen, S.C., 2002. An evaluation of the minimum-jerk and minimum torque-change principles at the path, trajectory, and movement-cost levels. *Motor control*, 6(1), pp.69–83.
- Kobayashi, K. & Hiraishi, K., 2012. Self-triggered model predictive control with delay compensation for networked control systems. In *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*. IEEE, pp. 3200–3205.

- Kober, J., Oztop, E. & Peters, J., 2010. Reinforcement learning to adjust robot movements to new situations. *Learning*, pp.2650–2655.
- Kober, J. & Peters, J., 2009. Learning motor primitives for robotics. In *2009 IEEE International Conference on Robotics and Automation*. IEEE, pp. 2112–2118.
- Kober, J. & Peters, J., 2010a. Policy Search for Motor Primitives in Robotics. *Machine Learning*, 84(1-2), pp.1–8.
- Kober, J. & Peters, J., 2010b. Practical Algorithms for Motor Primitives in Robotics. *Robotics*, 17(2), pp.1–8.
- Korb, K.B., 1998. The frame problem: An AI fairy tale. *Minds and Machines*, 8(3), pp.317–351.
- Kutch, J. et al., 2008. Endpoint Force Fluctuations Reveal Flexible Rather Than Synergistic Patterns of Muscle Cooperation. *Journal of Neurophysiology*, 100(5), pp.2455–2471.
- Kwon, W., Bruckstein, A. & Kailath, T., 1982. Stabilizing state-feedback design via the moving horizon method. In *21st IEEE Conference on Decision and Control*. IEEE, pp. 234–239.
- Lackner, J. & Dizio, P., 1994. Rapid adaptation to Coriolis force perturbations of arm trajectory. *Journal of neurophysiology*, 72(1), pp.299–313.
- Ladd, A. & Kavraki, L., 2004. Fast Tree-Based Exploration of State Space for Robots with Dynamics. In *Algorithmic Foundations of Robotics VI*. Springer, STAR 17, pp. 297–312.
- Lapointe, N. et al., 2009. Specific role of dopamine D1 receptors in spinal network activation and rhythmic movement induction in vertebrates. *Journal of Physiology*, 587(Pt 7), pp.1499–511.
- Latombe, J., 1991. Robot Motion Planning. In B. Wah, ed. *Wiley Encyclopedia of Computer Science and Engineering*. Kluwer Academic Publishers, pp. 2439–2446.
- LaValle, S. & Kuffner, J., 2001. Randomized Kinodynamic Planning. *The International Journal of Robotics Research*, 20(5), pp.378–400.
- LaValle, S.M., 2006. *Planning Algorithms* S. M. Lavalle, ed., Cambridge: Cambridge University Press.
- Levine, W., 1996. *The Control Handbook* W. S. Levine, ed., CRC Press.

- Li, J. et al., 2008. Bayesian network modeling for discovering “dependent synergies” among muscles in reaching movements. *IEEE Transactions On Bio-Medical Engineering*, 55(1), pp.298–310.
- Lichtwark, G. & Barclay, C., 2010. The influence of tendon compliance on muscle power output and efficiency during cyclic contractions. *The Journal of experimental biology*, 213(5), pp.707–14.
- Lungarella, M. et al., 2003. Developmental robotics: a survey. *Connection Science*, 15(4), pp.151–190.
- Lungarella, M. & Sporns, O., 2006. Mapping Information Flow in Sensorimotor Networks K. Friston, ed. *PLoS Computational Biology*, 2(10), p.12.
- Luque, N. et al., 2011. Adaptive cerebellar spiking model embedded in the control loop: context switching and robustness against noise. *International Journal of Neural Systems*, 21(05), pp.385–401.
- Ma, S. & Feldman, A., 1995. Two functionally different synergies during arm reaching movements involving the trunk. *Journal of Neurophysiology*, 73(5), pp.2120–2122.
- Marques, H. et al., 2010. ECCE1: the first of a series of anthropomimetic musculoskeletal upper torsos. In *10th IEEE/RSJ International Conference on Humanoid Robots*. IEEE, pp. 391–396.
- Matthews, P., 1969. Evidence that the secondary as well as the primary endings of the muscle spindles may be responsible for the tonic stretch reflex of the decerebrate cat. *The Journal of Physiology*, 204(2), pp.365–393.
- Mayne, D. & Michalska, H., 1990. Receding horizon control of nonlinear systems. *IEEE Control Systems*, 31(3), pp.52–65.
- McGeer, T., 1990. Passive Dynamic Walking. *The International Journal of Robotics Research*, 9(2), pp.62–82.
- Mehta, B. & Schaal, S., 2002. Forward models in visuomotor control. *Journal of Neurophysiology*, 88(2), pp.942–53.
- Meyer, J., Husbands, P. & Harvey, I., 1998. Evolutionary robotics: A survey of applications and problems Philip Husbands & J.-A. Meyer, eds. *Evolutionary Robotics*, 1468(1994), pp.1–21.
- Miall, R. et al., 1993. Is the cerebellum a smith predictor? *Journal of Motor Behavior*, 25(3), pp.203–16.

- Miall, R., 1998. The cerebellum, predictive control and motor coordination. *Novartis Foundation Symposium*, 218, pp.272–84.
- Miall, R. & Wolpert, D., 1996. Forward Models for Physiological Motor Control. *Neural Networks*, 9(8), pp.1265–1279.
- Microsoft, 2013. The Microsoft Kinect Sensor. Available at: <http://www.microsoft.com/en-us/kinectforwindows/> [Accessed June 10, 2013].
- Mirtich, B. & Canny, J., 1995. Impulse-based simulation of rigid bodies. In *Proceedings of the 1995 symposium on Interactive 3D graphics - SI3D '95*. New York, New York, USA: ACM Press, p. 181–ff.
- Miyamoto, H. et al., 2004. TOPS (Task Optimization in the Presence of Signal-Dependent Noise) model. *Systems and Computers in Japan*, 35(11), pp.48–58.
- Mizuuchi, I. et al., 2007. An advanced musculoskeletal humanoid Kojiro. In *2007 7th IEEE-RAS International Conference on Humanoid Robots*. IEEE, pp. 294–299.
- Moore, A. & Atkeson, C., 1995. The parti-game algorithm for variable resolution reinforcement learning in multidimensional state-spaces. *Machine Learning*, 21(3), pp.199–233.
- Neptune, R., Clark, D. & Kautz, S., 2009. Modular control of human walking: a simulation study. *Journal of Biomechanics*, 42(9), pp.1282–1287.
- Nijhawan, R., 1994. Motion extrapolation in catching. *Nature*, 370(6487), pp.256–7.
- Peters, J. & Schaal, S., 2004. Learning Motor Primitives with Reinforcement Learning. In *11th Joint Symposium on Neural Computation*.
- Peters, J. & Schaal, S., 2008. Reinforcement learning of motor skills with policy gradients. *Neural Networks*, 21(4), pp.682–97.
- Peters, J., Vijayakumar, S. & Schaal, S., 2003. Reinforcement Learning for Humanoid Robotics. *Proceedings of the third IEEE/RAS international conference on humanoid robots*, 18(7), pp.1–20.
- Petreska, B. & Billard, A., 2009. Movement curvature planning through force field internal models. *Biological Cybernetics*, 100(5), pp.331–350.
- Pfeifer, R. & Bongard, J., 2007. *How the body shapes the way we think: a new view of intelligence*, MIT Press.

- Pfeifer, R. & Iida, F., 2005. Morphological computation: Connecting body, brain and environment. B. Sendhoff et al., eds. *Japanese Scientific Monthly*, 58(2), pp.48–54.
- Pfeifer, R., Lungarella, M. & Iida, F., 2007. Self-organization, embodiment, and biologically inspired robotics. *Science*, 318(5853), pp.1088–93.
- Potkonjak, V., Svetozarevic, B, et al., 2010. Biologically Inspired Control of a Compliant Anthropomorphic Robot. In *Proceedings of the 15th IASTED International Conference on Robotics and Applications*. ACTA Press Scientific, pp. 182–189.
- Potkonjak, V., Svetozarevic, Bratislav, et al., 2010. Control Of Compliant Anthropomorphic Robot Joint. *Symposium on Computational Geometric Methods in Multibody System Dynamics*, 8(1), pp.85–95.
- Radkha, K. & von Stryk, O., 2012. Human-Like Model-Based Motion Generation Combining Feedforward and Feedback Control for Musculoskeletal Robots. In *Proc. 7th Annual Dynamic Walking Conference 2012*.
- Roh, J., Cheung, V. & Bizzi, E., 2011. Modules in the brain stem and spinal cord underlying motor behaviors. *Journal of Neurophysiology*, 106(3), pp.1363–78.
- Rosenstein, M., Barto, A. & Van Emmerik, R., 2006. Learning at the level of synergies for a robot weightlifter. *Robotics and Autonomous Systems*, 54(8), pp.706–717.
- Rusu, R. et al., 2009. Real-time perception-guided motion planning for a personal robot. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 4245–4252.
- Schaal, S., 1999. Is imitation learning the route to humanoid robots? *Trends in Cognitive Sciences*, 3(6), pp.233–242.
- Schaal, S. et al., 2004. Learning movement primitives. In *International Symposium of Robotics Research*. Citeseer, pp. 1–10.
- Schaal, S., Ijspeert, A. & Billard, A., 2003. Computational approaches to motor learning by imitation. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 358(1431), pp.537–47.
- Scholz, D. et al., 2011. Bio-inspired motion control of the musculoskeletal BioBiped1 robot based on a learned inverse dynamics model. In *2011 11th IEEE/RSJ International Conference on Humanoid Robots*. IEEE, pp. 395–400.
- Schultz, W., 2002. Getting formal with dopamine and reward. *Neuron*, 36(2), pp.241–63.

- Schultz, W., 1998. Predictive reward signal of dopamine neurons. *Journal of Neurophysiology*, 80(1), pp.1–27.
- Shibata, T. & Schaal, S., 2001. Biomimetic gaze stabilization based on feedback-error-learning with nonparametric regression networks. *Neural Networks*, 14(2), pp.201–16.
- Shkolnik, A. & Tedrake, R., 2009. Path planning in 1000+ dimensions using a task-space Voronoi bias. In *2009 IEEE International Conference on Robotics and Automation*. IEEE, pp. 2061–2067.
- Smith, O., 1959. A controller to overcome dead time. *ISA Journal*, 6, pp.28–33.
- Snedecor, G.W. & Cochran, W.G., 1989. *Statistical Methods* 8th Editio., Iowa State University Press.
- Sontag, E., 1998. *Mathematical control theory: deterministic finite dimensional systems*, Springer-Verlag.
- Sperry, R.W., 1950. Neural basis of the spontaneous optokinetic response produced by visual inversion. *Journal of comparative and physiological psychology*, 43(6), pp.482–489.
- Sreenath, K., Park, H. & Grizzle, J., 2009. Embedding Active Force Control within the Compliant Hybrid Zero Dynamics to Achieve Stable, Fast Running on MABEL. *The International Journal of Robotics Research*, pp.1–26.
- Stephan, K. et al., 2002. Conscious and subconscious sensorimotor synchronization--prefrontal cortex and the influence of awareness. *NeuroImage*, 15(2), pp.345–52.
- Sucan, I. & Kavraki, L., 2008. Kinodynamic Motion Planning by Interior-Exterior Cell Exploration. *Cell*, 57, pp.1–16.
- Sutton, R. & Barto, A., 1998. *Reinforcement Learning, An Introduction*, MIT Press Cambridge, MA, USA.
- Suzuki, M. et al., 1996. Application of the minimum jerk model to formation of the trajectory of the centre of mass during multijoint limb movements. *Folia Primatologica*, 66(1-4), pp.240–252.
- Tanaka, H., Tai, M. & Qian, N., 2004. Different predictions by the minimum variance and minimum torque-change models on the skewness of movement velocity profiles. *Neural Computation*, 16(10), pp.2021–2040.

- Theodorou, E., Buchli, J. & Schaal, S., 2010. Reinforcement Learning of Motor Skills in High Dimensions : A Path Integral Approach. *Policy*, 2(3), pp.2397–2403.
- Ting, L. & Macpherson, J., 2005. A limited set of muscle synergies for force control during a postural task. *Journal of neurophysiology*, 93(1), pp.609–13.
- Ting, L. & McKay, J., 2007. Neuromechanics of muscle synergies for posture and movement. *Current opinion in neurobiology*, 17(6), pp.622–8.
- Todorov, E., 2004. Optimality principles in sensorimotor control. *Nature Neuroscience*, 7(9), pp.907–915.
- Todorov, E. & Ghahramani, Z., 2004. Analysis of the synergies underlying complex hand manipulation. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 6, pp.4637–40.
- Tresch, M., Cheung, V. & D'Avella, A., 2006. Matrix factorization algorithms for the identification of muscle synergies: evaluation on simulated and experimental data sets. *Journal of neurophysiology*, 95(4), pp.2199–212.
- Tresch, M. & Jarc, A., 2009. The case for and against muscle synergies. *Current Opinion in Neurobiology*, 19(6), pp.601–607.
- Tsianos, K., Sucas, I. & Kavraki, L., 2007. Sampling-Based Robot Motion Planning : Towards Realistic Applications. *Computer Science Review*, 1(1), pp.2–11.
- Urmson, C. & Simmons, R., 2003. Approaches for heuristically biasing RRT growth. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 1178–1183.
- Valencia, F. et al., 2011. Moving horizon estimator for measurement delay compensation in model predictive control schemes. In *IEEE Conference on Decision and Control and European Control Conference*. IEEE, pp. 6678–6683.
- Valero-Cuevas, F., Venkadesan, M. & Todorov, E., 2009. Structured Variability of Muscle Activations Supports the Minimal Intervention Principle of Motor Control. *Journal of Neurophysiology*, 102(1), pp.59–68.
- Vanderborght, B. et al., 2004. LUCY, a Bipedal Walking Robot with Pneumatic Artificial Muscles. *Mechatronics*.
- Verrel, J., Lövdén, M. & Lindenberger, U., 2010. Normal aging reduces motor synergies in manual pointing. *Neurobiology of Aging*, 33(1), pp.1–9.

- Wan, E. & Van Der Merwe, R., 2001. The Unscented Kalman Filter S. Haykin, ed. *Kalman Filtering and Neural Networks*, 5(1), pp.221–280.
- Webb, B., 2004. Neural mechanisms for prediction: do insects have forward models? *Trends in neurosciences*, 27(5), pp.278–82.
- Weiss, E.J. & Flanders, M., 2004. Muscular and postural synergies of the human hand. *Journal of Neurophysiology*, 92(1), pp.523–35.
- Welch, G. & Bishop, G., 2006. An Introduction to the Kalman Filter A.-W. Acm Press, ed. *In Practice*, 7(1), pp.1–16.
- Wittmeier, S. et al., 2012. Calibration of a physics-based model of an anthropomimetic robot using Evolution Strategies. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 445–450.
- Wittmeier, S., Jantsch, M., et al., 2011. CALIPER: A universal robot simulation framework for tendon-driven robots. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 1063–1068.
- Wittmeier, S., Jantsch, M., et al., 2011. Physics-based Modeling of an Anthropomimetic Robot. In *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*. pp. 4148–4153.
- Wittmeier, S. et al., 2013. Toward anthropomimetic robotics: development, simulation, and control of a musculoskeletal torso. *Artificial life*, 19(1), pp.171–93.
- Wolpert, D., Ghahramani, Z. & Flanagan, J., 2001. Perspectives and problems in motor learning. *Trends in Cognitive Sciences*, 5(11), pp.487–494.
- Wolpert, D., Ghahramani, Z. & Jordan, M., 1995. An internal model for sensorimotor integration. *Science*, 269(5232), pp.1880–1882.
- Wolpert, D. & Kawato, M., 1998. Multiple paired forward and inverse models for motor control. *Neural Networks*, 11(7-8), pp.1317–1329.
- Wolpert, D., Miall, R. & Kawato, M., 1998. Internal models in the cerebellum. *Trends in Cognitive Sciences*, 2(9), pp.338–347.

**Chapter 9 :**  
**Appendices**

---

## **9.1 Appendix I: Physics engine comparison report**

This tabular comparison report was produced by the ECCERobot team as a resource for selection of the most appropriate physics based simulation software. From this report the Bullet Physics engine was selected for use in modelling the ECCERobot.