Observe, Predict, Adapt: A neural model of adaptive motor control

University of Waterloo

Natarajan Vaidyanathan

July 20, 2023

Contents

Abstract

Biological control systems have evolved to perform efficiently in an environment characterized by high uncertainty and unexpected disturbances, while relying on noisy sensors and unreliable actuators. Despite these challenges, biological control systems remain superior to engineered control systems in many respects. This edge in performance can be attributed to the exceptional ability of the brain to predict adaptively, and continuously update its control strategies in the face of uncertainties. Consequently, to harness these control abilities, it is crucial to delve into the study and modeling of cortical functioning.

This thesis presents a novel and comprehensive approach to elucidate the underlying mechanisms governing motor control. Specifically, we propose a biologically plausible spiking neural network model of the primate sensory-motor control system. The core of the model lies in its effective handling of noisy observations, integration of different sensory modalities, and the ability to learn to control the arm in the presence of perturbations. This is accomplished through the introduction of the Neural Adaptive Filter, a mechanism that dynamically predicts sensory consequences based on control inputs and observations.

The developed functional model of the sensory-motor control system exhibits complex behaviors observed in primates' reaching and demonstrates neural activities comparable to experimental findings. By adopting a spiking architecture, and connecting the lower-level synaptic dynamics and higher-level behaviors, such as visuomotor rotation, the model offers valuable insights into underlying mechanisms. Furthermore, the incorporation of anatomical structure and neural constraints enhances the biological plausibility and explanatory power of the model.

Moreover, the realization of a functional spiking model of the sensory-motor control system hols broader implications, particularly in control theory and its applications. This spiking model preserves the brain's inherent sparse coding, optimal performance, and energy efficiency, which are highly advantageous for engineering solutions. The model's operation can be generalized to a of a filter-controller framework capable of adapting to unknown nonlinear systems, enhancing robustness and plasticity derived from biological inspiration. Ultimately, by integrating the working of a biological control system into modern control theory, our model not only deepens the understanding of the working of sensory-motor control system, but also has far-reaching implications in advanced control methodologies.

Chapter 1

Introduction

Modern control theory and artificial neural network researchers have developed innovative algorithms to control various systems, ranging from robotic arms to spacecrafts. In the process of solving these diverse problems, better sensors, precise actuators and novel optimization techniques have emerged, but so has the demand for faster computations, energy efficiency and more sophisticated algorithms. In order to meet these demands, a controller is typically tailor-made for a particular problem. As a result, such controllers are typically highly specific, designed using a variety of particular control methods, and highly tuned. This often results in controllers with very little ability to adapt to dynamic situations.

When compared to modern technologies, biological systems outperform them at core control tasks with remarkable efficiency and speed. This superior performance is primarily due to biology's unique ability to process and integrate vast amounts of information simultaneously to adapt to changing conditions, and learn and evolve over time. For instance, consider the problem of autonomous navigation, one of the difficult challenges faced by engineers. Biology has not only solved this problem but has also demonstrated remarkable proficiency, even in the most primitive of brains. *Megalopta genalis*, a variety of bee, achieves its feat of rainforest navigation in the dead of night by estimating its location with fewer than five photons [Warrant, 2004]. Similarly, an essential element in the behavioral repertoire of a dragonfly is its swift flight. Dragonflies can skillfully maneuver, anticipate the trajectories of their prey, and intercept at exceptionally high velocities relative to its body length. Its small brain of only a million neurons is able to achieve this with a success rate as high as 97% [Olberg et al., 2005].

When considering neurobiological systems as complex as humans and other non-human primates, we find extraordinary learning, adaptability and dexterity in performing complex, context-dependant control tasks. What is more remarkable is the ability of the brain to exert control, and simultaneously adapt to the changes in its uncertain environment on-the-fly. The brain's ability to gracefully handle constantly changing internal goals, different external rewards, dynamic and kinematic transformations, and varying biological costs, further bolsters the claim that the brain remains superior in control to our current technologies. The advancement of neural sensory and motor systems through years of evolution has rendered them highly intricate and notoriously resilient. Thus, the sensory motor control system has the potential to help us improve our engineering control methodologies.

Understanding the brain's working and creating a model of its mechanisms is not a novel idea. Years of studying the cortex has provided a rich literature describing how different brain circuits and their mechanisms work. In fact, experiments from multiple domains of neuroscience has increased our breadth of understanding of the brain's working at varying functional levels [Sejnowski et al., 2014]. At a behavioral level, experiments have investigated how the brain chooses between different courses of action and how different functional areas of the cortex influence these decisions. Behavioral studies allows us to investigate the abilities of the brain to learns and adapt to a variety of tasks, but these studies are often limited in the information they provide about individual mechanisms that give rise to these behaviors. Experiments on a systems level, are better equipped to give insights into the relevant cortical pathways. Observations from studying injuries, disorders, and clinical experiments, can help us deduce the system states, and the functions of specific regions, and the relevant cortical circuits at a mechanistic level. On the other hand, with the help of multi-electrode arrays and optogenetic techniques, it is possible to dive further into the firing patterns, and how the cortical mechanisms manifest at a cellular level. Interestingly, experiments at multiple levels of the hierarchy, solve different parts of the same puzzle of how does the brain works. Hence, it is difficult for us to restrict ourselves to a single methodology to get a comprehensive understanding of the working of the brain. Rather, there is a need to combine findings to have a unifying model that is capable of explaining the workings from the synaptic level to higher level behavior.

To fuse the breadth of experimental findings to produce a unified computational brain model, we need to address some significant challenges. The fig.1 depicts how different experimental techniques can probes the brain at varying spatial and temporal resolutions [Sejnowski et al., 2014]. Recordings range from patch clamps and EEGs to large scaled fMRI and lesion studies. To approximate the neural mechanisms, it is necessary for our model to capture the details of synaptic level dynamics that happens within a few milliseconds as well as the brain's behaviors that spans across minutes or even longer timescales. It is important to note that many of the higher level behaviors are often dictated by the lower level circuitry. For instance, consider long-term potentiation (LTP) and long-term depression (LTD), [RobertC et al., 1999] forms of synaptic plasticity referring to the ability of synapses to change their strength in response to patterns of neural activity. Studies have shown how LTP in the hippocampus is critical for encoding and storing spatial information [Bliss and Collingridge,

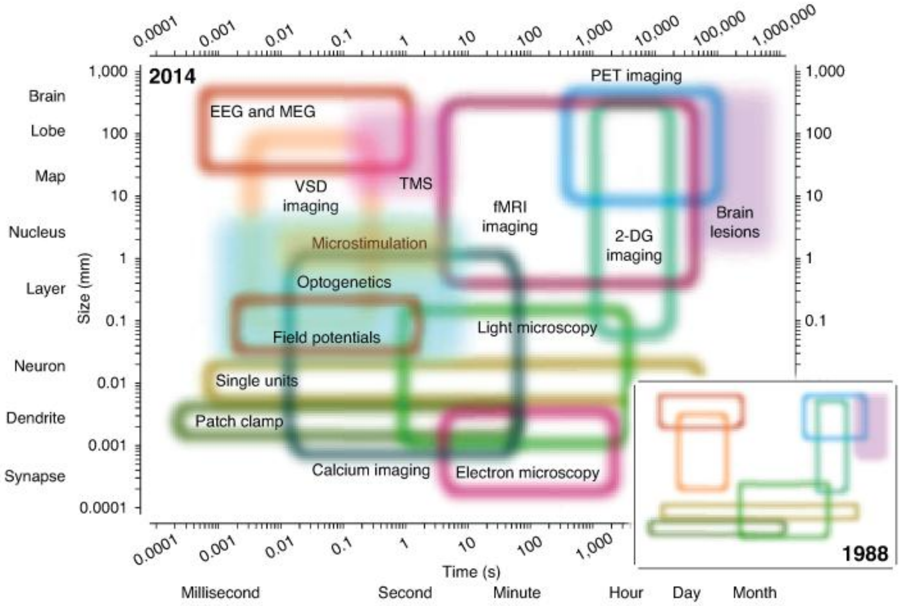

Figure 1.1: **The spatio-temporal domain of neuroscience methods available for the study of the nervous system as described in [Sejnowski et al., 2014]** . Each colored region represents the useful domain of spatial and temporal resolution for one method available for the study of the brain. Open regions represent measurement techniques; filled regions, perturbation techniques.

1993]. Also, particularly for motor control, LTD in the Purkinjee cells of cerebellum has shown to be is involved in motor learning and coordination [Masetty et al., 1989]. Changes in the connectivity between the amygdala and the prefrontal cortex can lead to changes in emotional regulation and decision-making, showing how a lower level change can dictate a higher level behavior. This evidence collectively supports the idea that it is necessary to model the brain at different scales to encapsulate the overall working of the brain, with comparable firing patterns, neural structures and synaptic connectivity.

While modelling the brain to exhibit organizational details captured by various levels of analysis, we should also remember to attend to the inherent biological constraints of the relevant mechanisms. For instance, we can model a neural network to perform a task of image classification, and use a very common optimization algorithm in Artificial Neural Networks (ANN's), such as back propagation. This neural network model, while capable of performing classification with remarkable accuracy [Lu and Weng, 2007], may not not guarantee biological realism. As shown by many, back propagation is not biologically realistic [Scellier and Bengio, 2017], hence makes the algorithm difficult to realize biologically and may not mimic the actual working of the brain. Likewise, it is possible to design a number of computationally adept models to solve a particular task, but it is the models with biological realism and comparable structure and function that are the ones most relevant for understanding the mechanisms of the brain. The model must be explainable and comparable with observed neuroscientific results, since the goal is to match the computational function of the cortex and not the mere performance at a given task. At first, the above additional constraints might appear as an unnecessary overhead to an already difficult problem of engineering the nuances of motor control. Interestingly, biological mimicry in modelling sometimes offers its own advantages. Constraining biological structure in encoding and information processing can help us incorporate effective computational strategies into our models. For example, in models that process images, V1 like cells emerge in convolutional models of vision, where the convolution is inspired by neural observations. Similarly,in [Cueva and Wei, 2018], the authors show grid-like representations, similar to those observed in the entorhinal cortex of rodents, emerge in artificial neural networks trained to perform spatial localization tasks. These ideas highlights a particular challenge in creating a unified model of the cortex. It is crucial to not merely reduce the brain's functions into to a black box, but to incorporate explainable structures with added neuro-anatomical constraints to not only have a comparable and comprehensive understanding of the brain, but also to enhance those solutions that we seek for our control problems.

While tackling the challenges in the modelling process, it is also important to select the system to model wisely – after all modelling the entire brain is currently an impossible task. The sensory motor control system is an exceptional candidate for modelling, however, not only for the control engineering implications but from a neurobiological perspective as well. Motor control involves the execution of coordinated movements and is relatively well understood compared to other complex cognitive processes such as memory or decision-making. Movements are observable and quantifiable behavior and makes it easier to model and compare against experimental findings. Furthermore, it is through the sensory and motor systems we interact with our reality. We experience sensory observations, anticipate changes and model our body as well as the environment and constantly adapt to environmental changes. In fact, the goals of the sensory motor control system such as handling uncertainties, predicting action outcomes and adapting to changes, are not objectives exclusive to motor control alone, but common to many other cortical systems. Given that there are strong parallels between other cortical systems and the motor control system, modelling the sensory motor system is an excellent target for improving our understanding of the brain. Furthermore, the engineering and control implications, generalizability to other perceptual systems, and applications in robotics and medical interventions all emphasize the necessity of such modeling endeavors. Consequently, modelling the sensory-motor control is a useful approach not only to enhance our engineering methodologies but also for building a more comprehensive model of the brain.

Currently, there are very few models of sensory-motor control that tie lower level dynamics with higher level behavior. One pioneering model that addresses this issue is the REACH model (Recurrent Error-driven Adaptive Control Hierarchy ([DeWolf et al., 2016]). The REACH model establishes connections between spiking neural networks and higher-level behaviors, incorporating experimental findings. To achieve this, the REACH model utilizes the Neural Engineering Framework (NEF), a tool for implementing neurobiological circuitry and processing information through spike trains. By leveraging the NEF, the REACH model not only organizes and structures the motor control system more effectively but also provides insights into the functionalities of this complex system. In subsequent sections, we will delve into a more detailed examination of the REACH model and the NEF, exploring their respective implementations and contributions to motor control. This discussion helps to situate the current research and shows how we can leverage the NEF for building a new model of sensory-motor control.

In this thesis, we introduce a biologically plausible spiking neuron model of the sensory-motor control system, which integrates lower-level dynamics with higher-level behavior. We begin by identifying the challenges present in control and estimation methodologies and contrast them with biological observations that shed light on the brain's approaches to control, estimation, and adaptation. These challenges include handling noisy observations, integrating multiple sources of sensory information, and effectively controlling the arm for various tasks. Based on the analysis of these challenges, we construct a system that aims to address them effectively. Initially, we present a linear version of the model for estimation, followed by a more accurate spiking nonlinear version. The model's performance is evaluated by testing it in a well-studied experimental paradigm and comparing the results against experimental data. The proposed model effectively handles noisy observations, integrates multiple sources of sensory information to make predictions, and successfully controls the arm for various tasks. By adopting a spiking architecture with anatomical and neurobiological constraints, the model replicates the functioning of the elements sensory-motor system, allowing for comparisons across different spatio-temporal resolutions, from synaptic to behavioral levels. The incorporation of anatomical structure and neural constraints ensures the model's biological plausibility and explanatory power. Furthermore, translating these mechanisms into an engineering context not only addresses control challenges but also harnesses advantages from bio-mimicry, such as optimality, robustness, and energy efficiency. The presentation of this model aims to foster the integration of neuroscience and engineering, offering inspiration for novel, sophisticated control methodologies. We believe this model will contribute to the development of a comprehensive, explainable, comparable, and accurate modeling framework, thereby hopefully furthering our understanding of the mammilian brains.

Chapter 2

Engineering Background

Dynamic systems are systems that evolve with time. The mathematical modelling of dynamic systems is ubiquitous in engineering. Such modelling has diverse applications ranging from controlling the temperature of a room to predicting the weather of a city. Regardless of the application, it is of great importance to study the interaction of the system variables and how the inputs can be controlled to produce desired behavior in the system. The fundamental principle of control engineering is to characterize these processes, and design tools for studying and manipulating the behaviour of a system to drive the system to the desired state. In this section we describe different control theoretical concepts and leverage this a foundation towards the building of our model. In control engineering, the main components typically involved in designing and implementing control systems are (see Fig.2.1):

- **Plant or dynamical system**: The plant refers to the physical system or process being controlled. In general, this could be a mechanical system, an electrical circuit, a chemical process, or any other system that exhibits dynamic behavior.
- **Sensors**: Sensors are devices that measure the relevant system signals of the plant. They provide feedback to the control system by converting physical quantities (e.g., position, temperature, pressure) into electrical signals that can be processed by the controller. In reality, it is often common to see noises added to the sensory measurements.
- **Estimator**: The estimator gathers the noisy measurements of the relevant system states from the sensors. Often combining multiple sensory information, the estimator provides an accurate estimate the current system state. This estimate is then compared with the desired system states to generate an error signal.
- **Controller**: The controller is responsible for determining the command signal. It takes the error signal as input and generates the control signal as output. The control signal is then sent to the actuator.

- **Actuators**: The calculated control signal influences the plant with the help of an actuator. An actuator converts the control signal into a physical action which could be a motor, a heater, a valve, or any other device that can manipulate the plant. Actuators, commonly physical devices, are subject to noises and disturbances that are capable of adding uncertainties into the system states.

Drawing parallels between the panels (a) and (b) of Figure.2.1, we can see how the classical control formulation can be helpful in modelling the sensory-motor control system. Essentially, in sensory-motor control, the system we aim at controlling is the arm interacting with the environment. We observe the system states using our vision (position and velocity information of the arm in Cartesian space) and proprioception (angular position and velocity information from the joint tension). It is our perception system that takes the role of an estimator in our control system. It uses the efference copy of the command signal and the two streams of noisy sensor information, to produce estimates in the two modalities and then the combined estimate of the hand location. This estimate is then compared with the desired trajectory of the arm to produce an error signal to generate the necessary command signal to follow the desired trajectory. Our muscles convert the neuronal activation into joint torques, completing the control loop that helps us move our hand to the desired trajectory. The added noise in neural transmission, physical disturbances such as gravity, external forces, etc. adds noise to the system as shown in the model in the Figure.2.1, (b). It is also important to note that there are details omitted here: for e.g., system identification / learning that occurs in biological estimation or adaptive control while moving the actuators. We shall discuss such subtleties and the dynamic abilities of the individual components in the later chapters. For now, we delineate the individual elements that define the working process of the control system we aim at modelling, and importantly, how these elements can be realized within the framework of control theory.

In the remaining sections of the chapter, we provide brief descriptions of control and estimation – moving from linear to nonlinear methodologies – and also discuss the current state of system identification to provide an engineering foundation for our later neurobiological modelling.

2.1 Linear control

Building control systems, historically started with linear controllers and estimators for linear systems. Linear Time Invariant (or LTI) systems are a class of systems whose outputs for a linear combination of inputs are the same as a linear combination of individual responses to those inputs. When the input/output relation does not change with changes in time, it refers to the time invariant quality of the system. When the system under control is an LTI system, we can employ a corresponding linear controller to move the system to the desired target optimally. An LTI system can be expressed as a set of n coupled first-order ordinary differential equations, known as the *state equations* (eq. 2.1). For

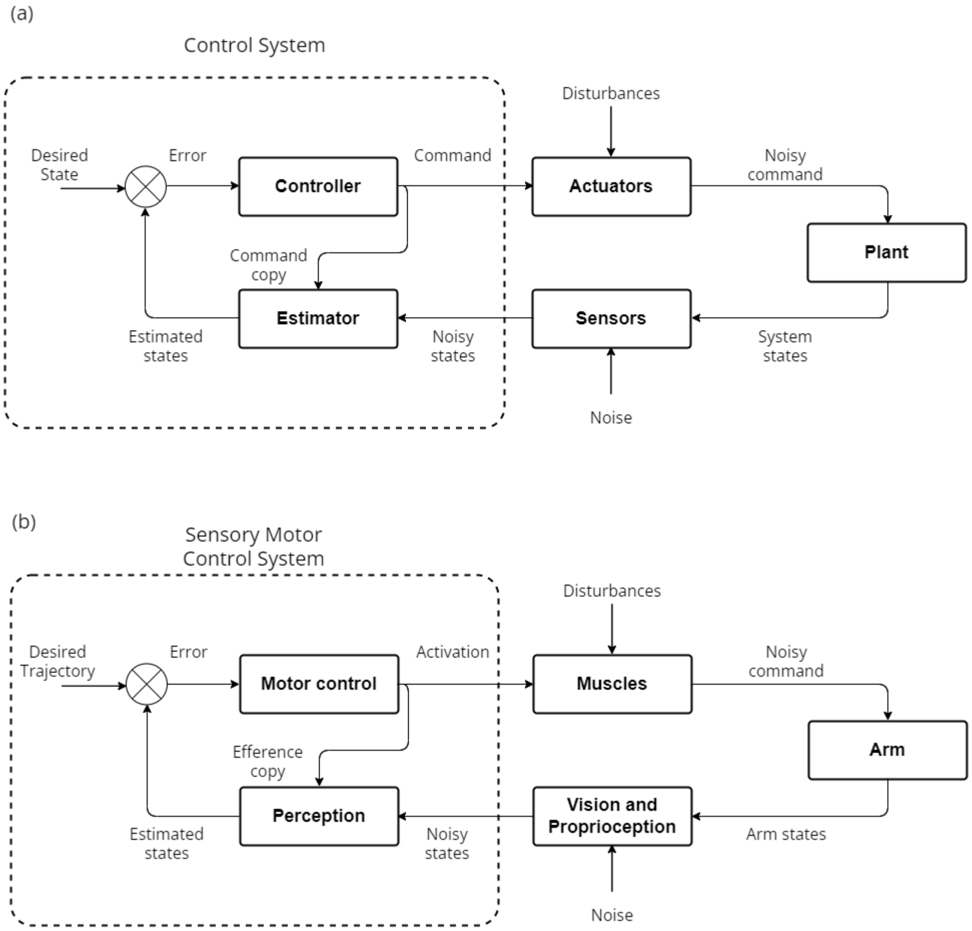

Figure 2.1: **State flow diagram of control feedback in classical control and sensory motor control.** The top panel (a) shows the flow of control states in a classical feedback control loop from the plant enabled by the sensors and actuators. The bottom panel (b) shows the comparison of control flow in a sensory motor feedback loop states flowing from the arm to the cortex through vision and proprioception and the muscle actuating the arm.

describing a linear system, the states are combined into a vector form $x \in \mathbb{R}^n$, propagated by the dynamics matrix $A \in \mathbb{R}^{n \times n}$ and the input matrix $B \in \mathbb{R}^{n \times m}$ for the control inputs $u \in \mathbb{R}^m$:

$$\dot{x}(t) = A\,x(t) + B\,u(t) \tag{2.1}$$

The above equation characterizes the evolution of the linear system. For the time invariant case, the A and B matrices do not change through time, and we have dropped the time notation for simplicity. Linear control theory revolves around optimally estimating the evolving system and driving the system states to the desired target.

The optimal linear control problem is solved by a quadratic cost for driving the states to zero [Stengel, 1994], [Kirk, 2004]. The cost function J is defined as:

$$J = \int_o^{t_f} \left(\frac{1}{2}\boldsymbol{x}^T Q x \, dx \; + \; \frac{1}{2} u^T R \, u \, du \right) \tag{2.2}$$

where Q and R are state and control penalties respectively. Solving for the above cost function using the *Algebraic Riccati equation*, we get the optimal gain for u to drive the $x \to 0$ as $t \to t_f$. Adding the optimal feedback control gain $K_c \in \mathbb{R}^{n \times m}$ we get the control law:

$$u^* = -K_c \varepsilon \tag{2.3}$$

$$\boxed{\dot{x} = (A - BK_c)\varepsilon} \tag{2.4}$$

By employing the above control, we ensure the system is driven to ($\varepsilon \to 0$). For controlling the system to a desired state x_{des}, the optimal control law becomes eq. 2.3, where the error is given by $\varepsilon = x_{des} - x$. Substituting it into 2.1 gives us the new system dynamics 2.4, and with the gain parameter K_c, where $K_c > 0$, we can change the eigen values of the combined matrix $A - BK_c$, making it negative, converging the system to be asymptotically stable at zero. The Figure 2.2 (a) shows the control diagram of the linear state space control problem. Since this setup regulates the system states to a desired target by optimally reducing a quadratic cost, this control structure is known as a *Linear Quadratic Regulator*. This method, if incorporated with an integral term for the error, becomes similar to the working of a control strategy often known as the *PID control*, and is a common controlling strategy in implementing a linear control system with a simple feedback loop.

2.2 Linear filter

The above control scenario, with a simple feedback gain, works only when we have access to the actual state of the system. Often we have limited access to the full state of the system, either due to uncertainty or system restrictions.

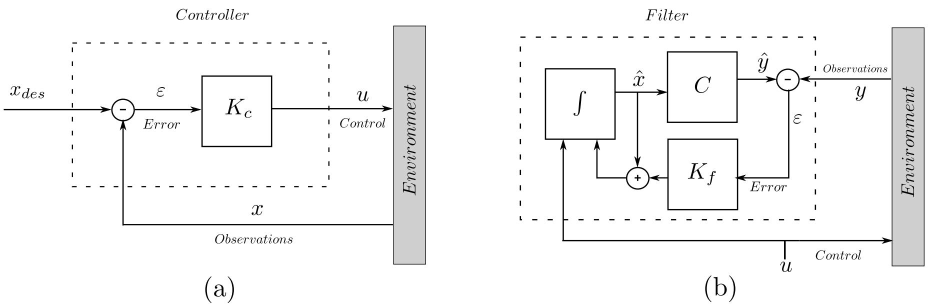

Figure 2.2: **Individual components of a Linear controller and a Linear filter**: (a) Shows the flow diagram of a linear controller giving the control signal u when the desired state x_{des} is given where K_c is the control gain. (b) Shows the flow diagram of a Linear filter or a Kalman filter, estimating $\hat{x}$ when given y, the observations are made from the environment. K_c is the controller gain and K_f is the filter gain

For example, we have access to the states of our arm through the noisy sensors of vision and proprioception. In such cases, it is necessary that we estimate the system states using our observations. An optimal linear filter or estimator provides a state estimate $\hat{x}$ by observing the noisy sensory information y and can be described as a form of linear observer:

$$\begin{aligned} \hat{y} &= C\hat{x} \\ \dot{\hat{x}} &= A\hat{x} + B\,u + K_f(y - \hat{y}) \\ \dot{\hat{x}} &= (A - K_f C)\hat{x} + \begin{bmatrix} B & K_f \end{bmatrix} \begin{bmatrix} u \\ y \end{bmatrix} \end{aligned} \tag{2.5}$$

The K_f is known as the optimal filter gain. We can determine how the error (ε) between the estimate and actual state is driven to zero. Let us combine our system equation 2.1 and the estimate equation 2.5 to get the propagation of our error.

$$\begin{aligned} \dot{\varepsilon} = \dot{x} - \dot{\hat{x}} &= (A\,x + B\,u) - (A - K_f C)\hat{x} + \begin{bmatrix} B & K_f \end{bmatrix} \begin{bmatrix} u \\ y \end{bmatrix} \\ &= A\,x - A\hat{x} + \cancel{B\,u} + K_f C\hat{x} - K_f\,C\hat{x} - \cancel{B\,u} \end{aligned} \tag{2.6}$$

$$\boxed{\dot{\varepsilon} = (A - K_f C)\varepsilon} \tag{2.7}$$

From the above equation, we can choose an appropriate K_f to make the combined $A - K_f C$ matrix negative, thereby ensuring the error converges to zero. This implies that the estimate $\hat{x}$ would eventually converge to the actual state x. Figure 2.2 (b) shows the flow diagram of the implementation of this linear filter formulation.

The optimal filter gain can be seen as reducing the quadratic cost function:

$$J = \int_{o}^{t_f} \left(\frac{1}{2} (x - \hat{x})^T Q \, (x - \hat{x}) \, dx \right) \tag{2.8}$$

where Q is the penalty term for the difference between the actual state and the estimated state. It is because of solving for the quadratic cost function, this formulation of the filter eq. 2.7 is called as *Linear Quadratic Estimator*.

Comparing the previous eq. 2.4 and eq. 2.7 shows the duality of the control and the estimation problems. In both cases we are solving for a quadratic cost, to give a gain that guarantees convergence. Employing these two together, gives us the *Linear Quadratic Gaussian* controller that together estimates the system states and regulates the system towards the desired trajectory.

2.2.1 The Kalman filter

The *Kalman Filter* is a reformulation of the Linear Quadratic Estimator that helps us in dealing with process and measurement noise. In addition to the linear assumptions we made for the LQR, we also assume zero-mean and Gaussian distributions for the noises added to the system. This gives us the Kalman filter, a common, optimal methodology for linear state estimation that is widely used. With the added noise, the linear system propagation is slightly modified as below:

$$\dot{x} = A\,x + B\,u + Gw \tag{2.9}$$

$$y = Cx + v \tag{2.10}$$

where A is the dynamics matrix, B is the control matrix, G is the dynamics noise matrix and C is the observation matrix. y corresponds to the observations of the system states x. w_d is the process noise, sampled from a Gaussian, zero-mean and of variance Q_n is the measurement noise, similarly sampled from a zero-mean Gaussian distribution with covariance R_n.

The combined system equation with a linear controller and Kalman gains becomes:

$$\begin{bmatrix} \dot{x} \\ \dot{\varepsilon} \end{bmatrix} = \begin{bmatrix} (A - BK_c) & BK_c \\ 0 & (A - K_f C) \end{bmatrix} \begin{bmatrix} x \\ \varepsilon \end{bmatrix} + \begin{bmatrix} I & 0 \\ I & -K_f \end{bmatrix} \begin{bmatrix} w_d \\ w_n \end{bmatrix} \tag{2.11}$$

It is important to note that in the combined system, the state convergence is still dictated by the control and filter gains. The knowledge of noise and uncertainty variance help in identifying how far the estimate is from the ground truth.

The Kalman filter technique is often expressed as an iterative approach as described below. With the knowledge of dynamics and measurement covariance Q and R and an estimate covariance, P, we can deduce the steps as:

$$\dot{P} = AP + PA^T + GQG^T - PC^T R^{-1} CP^T \tag{2.12}$$

$$K = P(t)C^T R^{-1} \tag{2.13}$$

$$\dot{\hat{x}} = (A\hat{x} + Bu) + K(y - \hat{x}) \tag{2.14}$$

The estimation covariance is often of interest during estimation to know how certain we are of the estimates from the Q_n and R_n. This gives more information on the certainty especially if Q_n and R_n become~~s~~ time varying. Also, during times when the observations ~~is~~ not available or ha~~s~~ a large noise, the measurement uncertainty R is increased implying~~,~~ the current $\hat{x}$ should be a function of our prediction system rather than the observation.

The preceding formulations have provided a framework for understanding linear dynamical systems and have facilitated the design of optimal controller and estimator pairs for such cases. However, real-world systems, including the biological control of our arm, often exhibit nonlinear behavior. Nonlinear systems involve intricate relationships and nonlinearities that cannot be adequately described by linear models. Consequently, linear control and estimation techniques fall short in capturing the rich dynamics needed to accurately model nonlinear systems. Therefore, alternative approaches specifically designed to handle nonlinear systems are essential for effective control and estimation. In the following section, we will explore the setup of nonlinear systems and delve into the corresponding techniques for control and estimation. By doing so, we aim to tackle the challenges posed by nonlinear dynamics and provide insights into effective strategies for addressing them.

2.3 Nonlinear control

The behavior of linear systems through time is well defined. Change to the input to the system produces a proportional change in the system states. For an LTI, there are several ways to understand the system response to a given control input and its further evolution through time, using techniques including the impulse response, Laplace transforms, root locus, Nyquist stability criterion, etc. [Bhattacharyya et al., 2018]. As we have seen, there exist well-established control and estimation techniques that work with great generality. The prescription of linear controllers and estimation works for all classes of problems that can be fully defined in a LTI framework. However, linear formulations cannot address the complexities of nonlinear systems due to their inherent limitations in capturing nonlinear dynamics and interactions. Fundamentally, nonlinear systems are not characterized by superposition principle, which means that the relationships between the system variables are neither proportional nor additive. Furthermore, nonlinear systems are highly dissimilar from one another, and are difficult to categorize. Hence, a control regime that works for one system might not work for another. Similar to systems in nature, this class of problems is characterized by inherently nonlinear interaction of states. Nonlinear systems include those systems that are characterized by chaos, which means the system evolution is highly sensitive to minute changes in the initial conditions bringing drastically different state trajectories [Slotine and Li, 1987]. The system becomes more complex if it is also time variant. These differences from typical linear systems makes nonlinear systems often impossible to solve analytically and hence it is difficult to devise generalized systemic procedures to such systems. Hence, there is a need to develop new methods to test for stability and novel tools to design controllers.

A nonlinear model can be represented by an equation of the form Eq. 2.15, where f is a nonlinear function and $x \in \mathbb{R}^{1\times n}$ state vector. The number of states n is called the order of the system. The input u is the control signal that drives the system:

$$\dot{x} = f(x(t), u(t)) \tag{2.15}$$

It is necessary to emphasize that nonlinearity is inevitable when modelling real world control systems. Many of the forces present in natural systems, including coriolis forces, drag, damping and friction, behaves nonlinearly. Furthermore, during the linear control formulation we assumed that the dynamic system parameters are known, time invariant constants. In practice the system parameters vary with time and some times present us with no means to measure quantitatively. For instance, consider the system of an inverted pendulum. The system parameters such as mass, length, damping from the air resistance, and friction can be measured with great precision. However, these physical systems are susceptible to wear, tear and fatigue, and modelling dynamic friction, damping, and corialis forces becomes impractical. In these cases of nonlinearities, applying linear control solutions to nonlinear systems, even with linearization, comes with severe limitations [Morgan, 2015]. Overall, when dynamic systems are characterized by non-linearities, a separate formulation for characterizing and controlling these systems is required, which we shall discuss in the following sections.

2.3.1 Lyapunov's direct method

Before we start devising control methodologies, let us understand the stability of nonlinear systems. Essentially, understanding the stability and the equilibrium of the system can help us study the evolution of the states and can be used as precursors to estimating and controlling the plant of interest. Lyapunov's theory provides us with tools for determining the stability of an equilibrium point or a trajectory of a dynamical system. An equilibrium point, also known as a fixed point, is a state of the system where the state variables do not change over time. Stability describes whether the system returns to the equilibrium point or diverges away from it over time. Consider the dynamical system described before:

$$\dot{x} = f(x(t), u(t)) \tag{2.15}$$

where x represents the state variables of the system, t is time, and $f(x)$ is a function that determines the system's behavior. An equilibrium point is defined as a state x^* such that $f(x^*) = 0$.

If there exists a function $V(x)$ (referred to as a Lyapunov function) that satisfies the following conditions, then Lyapunov's direct method prescribes that there are three types of stability that can be inferred: local stability, local asymptotic stability, and global asymptotic stability. Informally,:

1. If the candidate function $V(x)$ is locally positive definite and the derivative is locally negative **semi-definite**, then the system is stable inside a given radius.

$$V(x) > 0 \ \forall x \in \mathcal{B} \tag{2.16}$$

$$\dot{V}(x) \leq 0 \forall x \in \mathcal{B} \tag{2.17}$$

for some neighborhood $\mathcal{B}$ of the equilibrium, the system is proven to be **stable**.

2. If the candidate function is locally positive definite and the derivative is locally negative **definite**, then the system will always converge to the equilibrium state starting inside a given radius.

$$V(x) > 0 \ \forall x \in \mathcal{B} \tag{2.18}$$

$$\dot{V}(x) < 0 \forall x \in \mathcal{B} \tag{2.19}$$

for some neighborhood $\mathcal{B}$ of the equilibrium, the system is proven to be **locally asymptotically stable**.

3. If the candidate function is globally positive definite and the derivative is globally negative definite, then regardless of initial state the system will always converge to equilibrium, and the system is proven to be **globally asymptotically stable**.

$$V(x) > 0 \ \forall x \in \mathcal{B} \tag{2.20}$$

$$|x| \to \text{inf} \implies V(x) \to \text{inf} \tag{2.21}$$

$$\dot{V}(x) < 0 \ \forall x \in \mathcal{B} \tag{2.22}$$

The Lyapunov's candidate function $V(x)$ can be seen as analogous to the energy of a system, and help us in finding points of equilibrium where the system converges upon dissipating energy. The idea here is: "if the system continues to dissipate energy, then it will eventually settle down to a state of minimal energy or an equilibrium point". Asymptotic stability is a very desirable feature of a system, that points us in the direction of formulating control as we have seen before in linear systems – driving the system to a "zero" state. But this can be difficult to prove with the above theorems, as often $V(x)$ is only negative semi-definite, rather than negative definite, and finding a Lyapunov function candidate with a negative definite derivative can be very difficult. To alleviate the difficulty of finding a candidate function with a strictly negative derivative, the invariant set theorem can be used. An invariant set is any set of states of a dynamical system where once the set has been entered, the system remains in that set. To put these theories into practice, let us implement an example using the above prescriptions.

2.3.2 Adaptive nonlinear control

It is necessary to reiterate the fact that nonlinear systems are quite different from one another, and hence to manipulate the system, generalized control frameworks are not available. Instead, each system requires its own stability consideration and control formulation. Therefore, in the following sections, we set up specific examples of nonlinear dynamics, identify system stability, and also go a step further by manipulating the system to follow a desired state trajectory.

Dampled pendulum dynamics

Let us analyze the stability of a nonlinear system of a damped pendulum dynamics [Slotine and Li, 1987] (see Figure 2.3. It has a simple non-linearity. The governing equation of the damped pendulum is given by:

$$J\,\ddot{q} + b\,\dot{q}|\dot{q}| + mgl\,sin(q) = u \tag{2.23}$$

where m, l, J are the mass, length and inertia of the pendulum respectively, b is the damping coefficient and the g is the acceleration due to gravity.

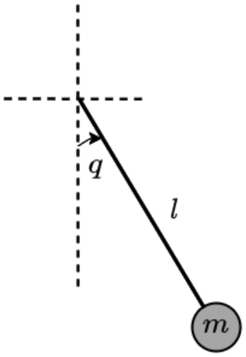

Figure 2.3: Damped pendulum model

Here we consider a trajectory control problem, where the desired trajectory to maintain is q_{des} and the tracking error is $\tilde{q}$:

$$\tilde{q} = q(t) - q_{des}(t) \tag{2.24}$$

For the ease of writing let us also introduce an intermediate variable q_r and a sliding variable s, given by :

$$\begin{aligned} s &= \dot{\tilde{q}} + \lambda\tilde{q} \ = \dot{q} - \dot{q}_r && (2.25) \\ \implies \dot{s} &= \ddot{q} - \ddot{q}_r && (2.26) \end{aligned}$$

where we define $\dot{q}_r = \dot{q}_{des} - \lambda\tilde{q}$. In having the prescriptions stated before, consider a Lyapunov candidate function $V(s(x))$ given by,

$$V(s) = \frac{1}{2} J\, s^2 \tag{2.27}$$

whose derivative now becomes:

$$\begin{aligned} \dot{V}(s) &= s\, J\, \dot{s} \ = \dot{s}(J\ddot{q} - J\,\ddot{q}_r) && (2.28) \\ &= s\,(u - b\dot{q}|\dot{q}| - mgl\, sin(q) - J\ddot{q}_r) && (2.29) \end{aligned}$$

Structuring this further, let us rearrange the system state variable separately from the system parameters θ_d. Then the equation 2.28 becomes:

$$\dot{V}(s) = s\,(u - Y_d(q, \dot{q}, \ddot{q}_r)\,\theta) \tag{2.30}$$

where $Y_d(q, \dot{q}, \ddot{q}_r) = [\ddot{q}_r \ \dot{q}|\dot{q}| \ sin(q)]$ and $\theta = [m \ b \ mgl]^T$. The Y_d is a set of known functions of the considered system states and the θ_d is the constant encompassing the system parameters.

If we were to apply the control law:

$$\boxed{u^*(t) = -K_s s + Y_d(q, \dot{q}, \ddot{q}_r)\,\theta} \tag{2.31}$$

Plugging it back into eq (2.28), then the function V becomes,

$$\dot{V}(s) = -K_s s^2 \leq 0 \tag{2.32}$$

According to Lyapunov's theory, the system has essentially become asymptotically stable, and since $u^*(t)$ penalizes the tracking error as a part of s, the system also follows the trajectory $q_{des}(t)$. As a result, we have successfully devised a control regime for a nonlinear system, with the help of Lyapunov's direct method.

It is important to draw a contrast between the classic linear control we have seen before and the new control regime implemented here. The choice of control here in eq 2.31, is not only a function of the system states, but now a function of the system parameters as well. This $u^*(t)$ driving the system on the desired trajectory hinges on the assumption that the system parameters, are known with certainty. However, there is uncertainty in the measurement of the system parameters and as a result this control choice can introduce noise in the system, the convergence of the system states cannot be guaranteed. To tackle this problem, we adopt the algorithms developed by Dr. Slotine ([Slotine and Li, 1987]). They provide an adaptive control algorithm to manage the system parameter uncertainties, thereby guaranteeing system convergence for a nonlinear system.

We shall demonstrate the new algorithm by continuing the previous example and its control regime. Let us consider an offset in system parameter measurements from the actual parameters $\tilde{\theta}$:

$$\tilde{\theta}(t) = \hat{\theta}(t) - \theta(t). \tag{2.33}$$

Given this new term, we are going to add this to the cost function.

$$V(s) = \frac{1}{2}Js^2 + \frac{1}{2}\tilde{\theta}^T P^{-1}\tilde{\theta} \tag{2.34}$$

where P^{-1} is a symmetric, positive semi definite matrix. The derivative can be calculated as:

$$\frac{d}{dt}\left(\frac{1}{2}\tilde{\theta}^T P^{-1}\right) = \frac{1}{2}\left(\dot{\theta} - \dot{\hat{\theta}}\right)^T P^{-1}\tilde{\theta} + \frac{1}{2}\tilde{\theta}^T P^{-1}\left(\dot{\theta} - \dot{\hat{\theta}}\right) \tag{2.35}$$

Given that $\dot{\theta} = 0$ indicating that the system parameters are constants, and that each constant term is equal to it's own transpose, the two terms become equivalent and the derivative can be rewritten as:

$$\frac{d}{dt}\left(\frac{1}{2}\tilde{\theta}^T P^{-1}\right) = -\dot{\hat{\theta}}^T P^{-1}\dot{\hat{\theta}} \tag{2.36}$$

Here, $\dot{\hat{\theta}}$ specifies how $\hat{\theta}$ changes over time. Hence it is sufficient for us to specify $\dot{\hat{\theta}}$ so that the newly added term cancels out, making the the system stable again. So, the reformulated adaptation law is given by:

$$\boxed{\dot{\hat{\theta}} = PY^T(q, \dot{q}, \ddot{q}_r)\, s} \tag{2.37}$$

We can see from the above eq.2.37, the sliding error of system states s also drives the system parameter learning too. Now the control law suggested from the previous section 2.31 combined with our eq. 2.37 provides an *Adaptive control* algorithm to control a plant to follow a desired state trajectory q_{des} while taking into account the uncertainty in the system parameters.

Two-link arm dynamics

This technique is useful in tackling a larger family of similar nonlinear dynamical systems. Next we consider an example of two-link arm dynamics (or a double pendulum dynamics) that is more relevant from a motor control perspective. We discuss how we can translate the working of the adaptive controller to this new dynamical system. The dynamics of the two-link model can be described by (see Figure 2.4:

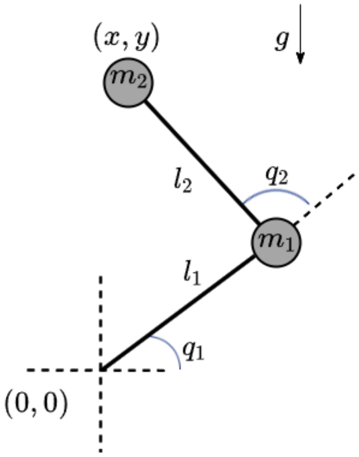

Figure 2.4: Two-link arm dynamical system

$$M\ddot{q} + C(q, \dot{q}) + g(q) = u \tag{2.38}$$

where M is the inertial matrix of the system in joint space, $C(q, \dot{q})$ is the matrix describing the Coriolis and the centrifugal component and the $g(q)$ is the term that describes gravity in joint space. Following a similar strategy as before, we can also define the sliding error and the reference trajectories. Now the $q(t)$ will essentially be the state vector containing both the joint's angles:

$$\tilde{q} = q(t) - q_{des}(t) \tag{2.39}$$

$$\begin{aligned} s = \dot{\tilde{q}} + \lambda\tilde{q} \ = \dot{q} - \dot{q}_r &\qquad (2.40)\\ \implies \dot{s} = \ddot{q} - \ddot{q}_r &\qquad (2.41) \end{aligned}$$

The candidate value function is:

$$V(s) = \frac{1}{2}s^T M(q)\, s, \tag{2.42}$$

and the derivative now becomes,

$$\begin{aligned} \dot{V}(s) &= s^T M(q)\,\dot{s} + \frac{1}{2}s^T\dot{M}(q)s &\qquad (2.43)\\ &= s^T M(q)\,\dot{s} + s^T C(q, \dot{q})s &\qquad (2.44)\\ &= \dot{s}\Big(M(q)\ddot{q} - M(q)\ddot{q}_r + C(q, \dot{q})\dot{q} - C(q, \dot{q})\dot{q}_r\Big) &\qquad (2.45)\\ \dot{V}(s) &= \dot{s}\Big(u - g(q) - M(q)\ddot{q}_r - C(q, \dot{q})\dot{q}_r\Big) &\qquad (2.46) \end{aligned}$$

Setting the control signal equal to:

$$\boxed{u = -K_s s + g(q) + M(q) q_r + C(q, \dot{q})\dot{q}_r} \tag{2.47}$$

where K_s is a symmetric positive definite matrix, and substituting into eq. 2.46 gives:

$$\dot{V}(s) = -s^T K_s s \leq 0. \tag{2.48}$$

The system is globally asymptotically stable, and now it is guaranteed to have zero position and velocity error when it enters the steady-state because $\dot{V}$ contains $s = \dot{q} - \dot{q}_r$, which must be zero when the system is at equilibrium.

Adaptation to unknown dynamics

The given control design above compensates for the gravity $g(q)$, inertial $M(q)$ and Corialis and centripetal forces $C(q, \dot{q})$ assuming that a perfect model of the system dynamics is known. As discussed earlier, this is often unrealistic and we need to adapt to the uncertainties to avoid adding error or otherwise the system becomes unstable. Similar to our previous system, we can split forcing function into unknown system parameters θ and the function of observable system states Y:

$$u = s^T K_s + Y(q, \dot{q}, \ddot{q}_r)\hat{\theta}. \tag{2.49}$$

The external forces that are not accounted for are thus counter-acted by learning the weights for a set of basis functions, $Y \in \mathbb{R}^{n \times d}$, defined over system states q and $\dot{q}$. The output of the basis functions is weighted by a set of learned parameters $\hat{\theta} \in \mathbb{R}^{d \times 1}$ consisting of estimates of $[\hat{M}(q) \big| \hat{C}(q, \dot{q}) \big|\ g(q)]^T$. Following the same method as before, choosing the candidate function becomes:

$$V(s) = \frac{1}{2} s^T M(q)\, s + \frac{1}{2}\tilde{\theta}^T L^{-1} \tilde{\theta} \tag{2.50}$$

The first term in the eq. 2.50, is similar to our previous equation 2.43. Hence let us calculate the derivative of the second term:

$$\frac{d}{dt}\Big(\frac{1}{2}\tilde{\theta}^T L^{-1}\tilde{\theta}\Big) = \frac{1}{2}(\dot{\theta} - \dot{\hat{\theta}})^T L^{-1}\tilde{\theta} + \frac{1}{2}\tilde{\theta}^T L^{-1}(\dot{\theta} - \dot{\hat{\theta}}) \tag{2.51}$$

$$\frac{d}{dt}\Big(\frac{1}{2}\tilde{\theta}^T L^{-1}\tilde{\theta}\Big) = -\dot{\hat{\theta}}^T L^{-1}\tilde{\theta} \tag{2.52}$$

To bring about stability in the system, consider the adaptation law:

$$\dot{\hat{\theta}} = L Y^T(q, \dot{q}, \ddot{q}_r) s, \tag{2.53}$$

Now, substituting this in the candidate function 2.51 the derivative becomes:

$$
\begin{aligned}
\dot{V}(s) &= -K_s s^2 + sY(q,\dot{q},\ddot{q}_r)\tilde{\theta} - \dot{\hat{\theta}}^T L^{-1}\tilde{\theta} \\
&= -K_s s^2 + sY(q,\dot{q},\ddot{q}_r)\tilde{\theta} - (LY^T(q,\dot{q},\ddot{q}_r)s)^T L^{-1}\tilde{\theta}, \\
&= -K_s s^2 + sY(q,\dot{q},\ddot{q}_r)\tilde{\theta} - sY(q,\dot{q},\ddot{q}_r)LL^{-1}\tilde{\theta}, \\
&= -K_s s^2 + sY(q,\dot{q},\ddot{q}_r)\tilde{\theta} - sY(q,\dot{q},\ddot{q}_r)\tilde{\theta}, \\
&= -K_s s^2 \leq 0.
\end{aligned}
$$

and the system is proven to be globally asymptotically stable. With the choice of the control law eq. 2.49, and the adaptation law in eq. 2.53, it it now guaranteed that the system will follow the trajectory given by $q_{des}, \dot{q}_{des}$, and more importantly converge to this trajectory even with imperfect measurement of the system states initially.

The above methods of Lyapunov and the derived adaptive control mechanisms help us address the limitations of traditional linear control methods in dealing with complex and highly nonlinear systems. In short, the Lyapunov method can help us analyze a nonlinear dynamical system and with careful choice of control we can drive the systems to the desired state trajectory.

2.4 Nonlinear filter

With a foundation in nonlinear control established, the subsequent sections will delve into the methods of nonlinear filter (or nonlinear estimators). Nonlinear estimation is as an integral component within a control loop, providing the essential state information necessary for guiding control actions by processing the noisy state observations. In the following sections we briefly examine the principles underlying two different nonlinear state estimation techniques, namely the Extended Kalman filters, and the particle filters.

2.4.1 Extended Kalman filter

The Extended Kalman filter (EKF) is a nonlinear state estimation technique that extends the traditional, previously seen linear Kalman Filter to handle systems with nonlinear dynamics. It is widely used in control systems, and signal processing, where accurate state estimation is crucial. Let us understand the process of state estimation of EKF.

The system equations are given by :

$$
x = f(x,u) + w_d \tag{2.54}
$$

$$
y = h(x) + w_n \tag{2.55}
$$

where $w_d \sim \mathcal{N}(0, Q(t))$ and $w_n \sim \mathcal{N}(0, R(t))$ are process and measurement noise, respectively. We initialize the estimated state $(\hat{x})$ and covariance (P) as,

$$
\hat{x}(t_0) = E[x(t_0)],\ P(t_0) = Var[x(t_0)] \tag{2.56}
$$

To update the prediction, we propagate both the state estimate and covariance:

$$\dot{\hat{x}} = f(\hat{x}, u) + K(y - h(\hat{x})) \tag{2.57}$$

$$\dot{P} = FP + PF^T - KHP + Q \tag{2.58}$$

$$K = PHR^{-1} \tag{2.59}$$

$$F = \frac{\partial f}{\partial x}|_{\hat{x},u} \tag{2.60}$$

$$H = \frac{\partial h}{\partial x}|_{\hat{x}} \tag{2.61}$$

where K is the *Kalman gain*, and the state transition and observation matrices are the Jacobians F and H. Note here that x, h, y, P, K, F and H are all functions of time, but the time notation is dropped for readability.

Here we handle the nonlinear state transition matrix by linearizing the system dynamics function and observation function. To propagate the estimated state and finding derivatives of the state transition function, we assume that we have perfect knowledge of the system dynamics and they do not change over time.

2.4.2 Particle filter

Another approach to estimating state propagation of a nonlinear system is the particle filter. A particle filter is a Bayessian state estimator that uses discrete particles to approximate the state estimate [Lundquist et al., 2015; Thrun, 2002]. The EKF, being a Gaussian filter, assumes a unimodal Gaussian distribution for the state estimate. In contrast, particle filters are capable of capturing and representing multiple modes in the state estimate.

The particle filter formulation, often described in discrete space, is given by:

$$x_k = f(x_{k-1}, u_k, v_{k-1}) \tag{2.62}$$

$$z_k = h(x_k, u_k, n_k) \tag{2.63}$$

Drawing parallels from eq. 2.15, the f is the dynamic function that maps the state x_{k-1} at time step $k-1$ to the next time step k and h is the observation function. A particle is an individual state estimate defined by its location in the state space $\hat{x}$ and a probability p, that indicates the likelihood of the estimation. The samples of a posterior distribution are called *particles* and are denoted by:

$$\chi_t = \{x_t^{[1]}, x_t^{[2]}, x_t^{[3]}, ..., x_t^{[M]}\} \tag{2.64}$$

Now the state estimation problem is addressed as a Bayessian estimation problem. The posterior distribution at the previous time step $p(x_{k-1}|z_{1:k-1})$, is combined with the process model that describes how the state evolves over time in the prediction step. The result is referred to as the prior state:

$$p(x_k|z_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|z_{1:k-1})dx_{k-1} \tag{2.65}$$

The *prior* represents the best guess at a given time, given measurements upto time $k-1$. The algorithm for implementing a particle filter is:

1. Sample $x_t^{[m]} \sim p(x_t|z_{1:t}, u_{1:t})$
2. Calculate $w_t^{[m]} = p(z_t|x_t^{[m]})$
3. Update $\chi_t^+ = \chi_t + < x_t^{[m]}, w_t^{[m]} >$
4. For all partcles, draw i with probability with $\propto w_t^{[m]}$
5. Add $x_t^{[i]}$ to χ_t

The denser a subregion of the state space is populated by samples, the more likely it is that the true state falls into this region.

Although the particle filter is a powerful estimation technique for nonlinear systems, especially effective at capturing noises from a multimodal distribution, they come with limitations. Firstly, this technique for solving for estimation using multiple particles is computationally expensive. It becomes difficult to scale as soon as we have many states in the system. Another limitation of a particle filter is that the algorithm is essentially nondeterministic. The fact that the particles are drawn with a sampling technique, could give rise to different state estimates depending upon the collective sample obtained.

Both the EKF and the particle filters help us control known nonlinear systems, estimate system states, and leverage system propagation to adapt to the varying unknown parameters of the system. This controller/estimator pair is also found in an approach called Model Predictive Control (MPC) where we observe the systems, predict the plant's system states for a time horizon, and produce the appropriate command to control the system of interest. Depending upon the system, the required control and the noise involved, a good choice for filtering can be made.

However, in designing MPC's, irrespective of the individual computational advantages with each estimation methods (eg. the EKF, the particle filter etc.), they collectively suffer from a common challenge. In propagating the system state estimates, it is often assumed that the dynamic system is constant, and more importantly, that it is known with certainty. In reality, system interactions change with time. From a motor control perspective, we have to manipulate different objects with varying dynamic and kinematic nonlinearities. Our brain is capable of handling new dynamics with different state interactions, and learns to control the new system. The current estimator techniques are not sufficient to handle unknown and changing nonlinear dynamics. This is where we employ the methods of system identification, which we shall discuss in the following section.

2.5 System identification methods

Here we discuss a scenario where we need to identify the interaction of the system states with one another, to employ predictive control methods. Given a continuous stream of observations, the problem is to deduce the dynamics of the plant. Techniques that solve this problem are often known as system identification or dynamics system discovery methods. These methods use statistical techniques to obtain mathematical models of the dynamics of interest, necessary both to predict the future, and if possible, drive the system in a desired trajectory. Such methods are employed in numerous applications across several disciplines. It is useful in modelling climate [Lim and Zohren, 2021], biology [MacNeil et al., 1992], finance [Lee et al., 2006]. Unsurprisingly, such methods also apply to modelling the motor system. Our brain tries to model the surroundings all the time. We observe different states spaces, across different time spans to model changes that happens around us – from dropping a ball to the change of seasons. While some predictions are more abstract, we here delve into system dynamics frameworks for a control system. It should be noted that these methods are not simply estimation techniques, but a system identification for a plant of unknown dynamics.

Historically, there have been several techniques that are explored for dealing with data driven system identification, including Dynamic Mode Decomposition (DMD) [Schmid, 2010], Koopman theory [Budišić et al., 2012] [Williams et al., 2015], nonlinear autoregressive models [Akaike, 1969], and neural network models [Wehmeyer and Noé, 2018] [Lu et al., 2021], [Raissi et al., 2019]. Effectively, system identification techniques can be classified into two different approaches:

1. **Black box modelling**: These methods often work without assuming any prior mathematical structure to the observed data. They try to fit the observation configuration given as much data as possible to constrain the model. Most system identification methods fall under this category. While system evolution can be predicted with some certainty, it is often difficult to explain how the system states actually interact to give rise to these dynamics.

2. **White box modelling**: These methods are inspired by the style of modelling that uses the first principles to impose structure on the observed data and fit a prioiri constraints to estimate the system model. Given these constraints, it is more explainable at multiple levels as to how the system states interact to propagate the dynamics

In the next section, we take a detailed look at one recent example of white box modelling that captures system dynamics called Sindy-C. This method provides a state-of-the-art benchmark for a new, biologically plausible method that we discuss later in the thesis.

2.5.1 Sparse Identification of nonlinear dynamics with control - Sindy-C

The sparse identification of nonlinear dynamics with control (SINDy-C) [Budišić et al., 2012; Kaiser et al., 2018] approach uses sparse regression methods to approximate the equations governing the system. Sindy-C trains on state observations to give a concise description of interactions between variables to describe the system. SINDY-C also enables us to sweep through the parameters to choose the Pareto-optimal model from a family of models by balancing accuracy and efficiency.

Let us derive the working of the SINDY-C algorithm. Consider a non-linear system described as:

$$\frac{d}{dt}x = f(x,u),\ x(0) = x_0. \tag{2.66}$$

~~Now~~ ~~g~~ive m measured snapshots of the state $x \in \mathbb{R}_{[p\times 1]}$ and the input signal $u \in \mathbb{R}_{[q\times 1]}$ in time, we arrange the observations into two matrices:

$$X = \begin{bmatrix} | & | & & | \\ x_1 & x_2 & ... & x_m \\ | & | & & | \end{bmatrix}_{p\times m} \qquad U = \begin{bmatrix} | & | & & | \\ u_1 & u_2 & ... & u_m \\ | & | & & | \end{bmatrix}_{q\times m}$$

The library of the candidate nonlinear functions can be chosen using the observed measurements X and input U, for example:

$$\Theta(X,U) = \left[1^T\, X^T\, U^T\, (X\otimes X)^T\, (X\otimes U)^T ...\, sin(X)^T\, sin(U)^T\, sin(X\otimes U)^T\, ...\right]$$

where ~~x~~$\otimes$~~y~~ defines the vector of all product combinations of the components in X and U. A suitable library of candidate terms is crucial for the working of SINDY-C algorithm. One potential strategy is to start with a common choice of polynomials, and increase the complexity of the library by including other nonlinear terms (trigonometric functions, exponential etc). ~~A p~~rior knowledge of the physics of the system of interest can also help us in making a smarter choices of functions ~~too.~~

The system equation can thus be written as:

$$\dot{X} = \Xi\Theta^T(X,U) \tag{2.67}$$

The time derivative $\dot{X}$, if not measured directly can be computed by numerical differentiation. With the new formulation, we employ sparse regression to identify a Ξ. This regression is done by balancing sparsity, which correspond~~ing~~ to fewest nonlinearities to describe the system, with the accuracy ~~to fit~~ the given measurements. The components ξ of Ξ are obtained using:

$$\xi_k = \arg\min_{\hat{\xi}_k} \frac{1}{2}\|\dot{X}_k - \hat{\xi}_k\theta^T(X,U)\|_2^2 + \lambda\|\hat{\xi}_k\|_1. \tag{2.68}$$

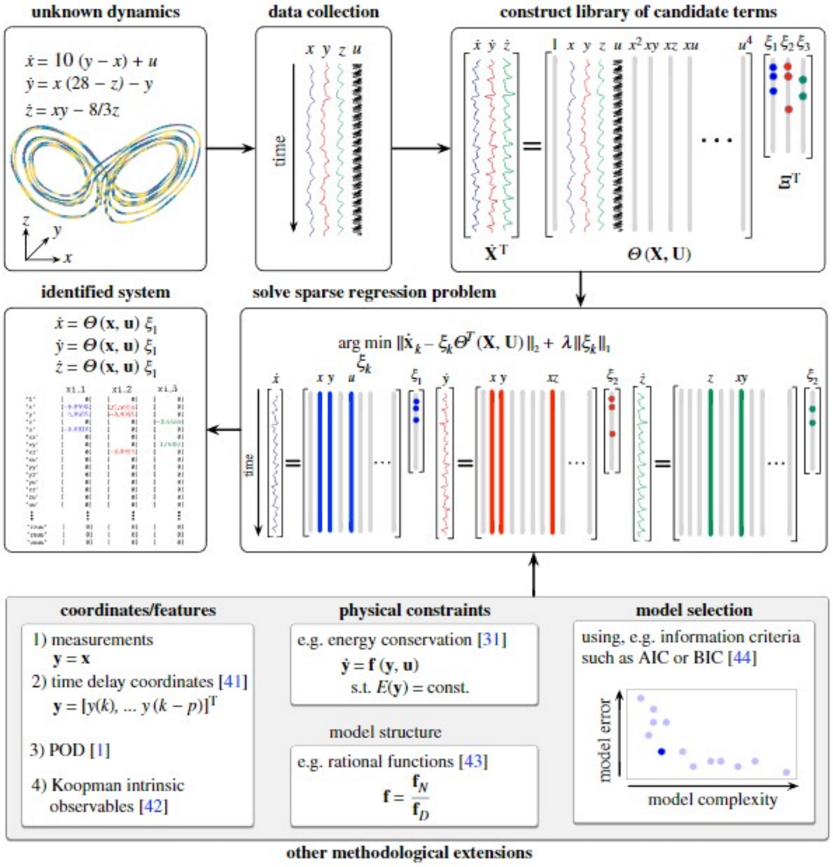
unknown dynamics
$\dot{x} = 10(y - x) + u$
$\dot{y} = x(28 - z) - y$
$\dot{z} = xy - 8/3z$
data collection
time
x y z u
construct library of candidate terms
$\dot{x}$ $\dot{y}$ $\dot{z}$
1 x y z u x^2 xy xz xu u^4
ξ_1 ξ_2 ξ_3
$\dot{\mathbf{X}}^T$
$\Theta(\mathbf{X}, \mathbf{U})$
Ξ^T
identified system
$\dot{x} = \Theta(\mathbf{x}, \mathbf{u})\,\xi_1$
$\dot{y} = \Theta(\mathbf{x}, \mathbf{u})\,\xi_1$
$\dot{z} = \Theta(\mathbf{x}, \mathbf{u})\,\xi_1$
solve sparse regression problem
$\arg\min_{\xi_k} \|\dot{\mathbf{x}}_k - \xi_k \Theta^T(\mathbf{X}, \mathbf{U})\|_2 + \lambda\|\xi_k\|_1$
coordinates/features
1) measurements $\mathbf{y} = \mathbf{x}$
2) time delay coordinates [41] $\mathbf{y} = [y(k), \ldots y(k-p)]^T$
3) POD [1]
4) Koopman intrinsic observables [42]
physical constraints
e.g. energy conservation [31]
$\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}, \mathbf{u})$
s.t. $E(\mathbf{y}) = \text{const.}$
model structure
e.g. rational functions [43]
$\mathbf{f} = \frac{\mathbf{f}_N}{\mathbf{f}_D}$
model selection
using, e.g. information criteria such as AIC or BIC [44]
model error
model complexity
other methodological extensions

Figure 2.5: **SINDY-C model**: Schematic of the SINDY-C algorithm and extensions.Active terms in a library of candidate nonlinearities are selected via sparse regression.Illustration of the modular nature of the SINDY-C with control framework(bottom row) and its ability to handle high-dimensional systems, limited measurements, known physical constraints and model selection.

where $\dot{X}_k$ represents the k^{th} row of $\dot{X}$ and ξ_k is the k^{th} row of Ξ. Upon regression, the Ξ can be used to describe the system equations as a linear combination of polynomials.

This method of sparse regression has many advantages. With very little data, we can approximate the system interaction and it is comparatively robust against noise [Budišić et al., 2012]. Unlike many blackbox methods, SINDY-C is also explainable. Despite these advantages, there are a few drawbacks to address in implementing this technique for modelling the sensory-motor control system of ~~our~~ brain~~.~~

- Choice of functional basis: SINDY-C is highly sensitive to the choice of functional bases ~~we choose~~ in Θ. Consequently, with a poor choice of the library of functions, an accurate system dynamics may not be ~~arrived.~~

- Cannot do feedback control: If the input u corresponds to feedback control, so that $u = f(x)$, it becomes impossible to disambiguate the effect of the feedback control u from the interactions within the dynamical system; In other words, the SINDY-C regression becomes ill-conditioned.

- No ~~learn~~ learning: SINDY-C is not a recurrent algorithm, but relies on a separate training and testing phases. Hence it is impossible to correct for errors online as we obtain new observations. This effectively prevents us from estimating ~~of~~ evolving systems or perform dynamic or kinematic learning, which are essential for model ~~our~~ biological control system.

- Incompatible biological framework: This framework inherently works with a process that cannot be reproduced using a biological system (library of polynomials, etc). Hence SINDY-C methodology is not apt for the purposes of ~~neuro-scientifically realistic~~ model of the brain.

- Numerical solving: After system identification, the system is solved for the state propagation using a numeric solver. This state trajectory is only as good as the choice of the numerical solver. Even when system is identified, since the dynamics are predominately chaotic, this method of system propagation can cause~~s~~ the system states to diverge from the ~~actual~~ states of the system.

This chapter has provided a comprehensive overview of various control theory concepts, encompassing linear control, linear filters, nonlinear controls, nonlinear filters, and system identification methods. While these techniques serve as valuable foundations for control engineering, it is important to recognize their limitations and challenges when applied to model real-world systems, particularly when the goal is to biologically realize perception and control systems. Real-world systems often involve nonlinearit~~y~~ and unfamiliar dynamics that cannot be adequately addressed by linear approaches alone. To address these challenges, it is necessary to build upon these fundamental concepts and leverage their strengths in developing more sophisticated models and techniques.

Bridging the gap between theory and practice requires considering the biological background of motor control systems within our brain. The upcoming section explores the existing body of knowledge and research, providing insights from relevant studies that inform our understanding and guide the construction of our model. By capturing the mechanisms and abilities of the motor control system and framing the problem from an engineering perspective, we can incorporate the wisdom gained from this broader context. This approach enables the development of more robust and accurate models for the complex motor control systems found in biological systems.

Chapter 3

Neuroscience background

Understanding the intricacies of primate sensory-motor control systems is essential in developing accurate models capable of effectively mimicking and manipulating such systems. ~~Drawing upon the engineering background of nonlinear control systems, we possess a solid foundation towards building a mathematical framework for the control feedback system.~~ By delving into the behavioral and neuroscientific aspects of motor control, we aim to obtain insights to enhance our understanding of mechanisms that drive ~~the~~ rich behaviors. ~~Significant contributions from research have~~ shed light on the vital roles played by various brain regions, such as the motor cortex, basal ganglia, and the cerebellum, in orchestrating and coordinating complex movements. Furthermore, the interplay between sensory input, motor commands, and feedback loops collectively helps ~~in to achieving~~ movement goals. By reviewing relevant studies and findings, we can uncover the ~~necessary~~ components required for ~~seamless~~ execution of motor tasks, and modelling the biological control system.

3.1 The ~~B~~rain: a powerful controller

Biological systems deal with highly non-linear and unreliable systems by observing the states through often noisy and delayed sensors. ~~They~~ also optimally integrate the sensory information to estimate the relevant system states and solve for the control required. In many ways, nonlinear control and estimation is a solved problem, and our brains have surpassed the performance of many of the current engineering solutions. In the following sections, we discuss ~~the~~ experimental findings and how they describe specific qualities of the cortex that are ~~cursory~~ for the efficient control of limbs.

3.1.1 Prediction

The delays in most sensorimotor loops are large. Efferent sensory signals are delayed as a result of neural conduction latency ~~and~~ low-pass filtering and interaction between neurons and muscles. It can take 100 ~~∼~~300 ms to send ~~the~~ command, move the arms and receive ~~the~~ visual feedback to perceive ~~the~~ change of hand location [Miall et al., 1993; Saunders and Knill, 2003; Miall et al., 2007]. The delays present in the sensory feedback loops are thus long enough to cause stability issues in the system ~~control.~~ Counter-intuitively, the stable control of the limbs indicates the presence of an internal mechanism to handle these delays ~~and~~ regulating movements, even before the controller receives ~~a~~ sensory feedback. In fact, [Colby et al., 1992] found evidence suggesting that prediction of the sensory consequence in a saccade movement study, can be seen explicitly in monkeys. A group of cells in posterior parietal cortex (PPC), fires in anticipation of a light stimulus falling on the receptive field as the animal moves the eye from one location to another. In another example, [Haarmeier et al., 1997] conducted a study related to the perception of motion on a patient ‘RW’. The patient, who suffered a stroke in the parietal and occipital cortex, reported seeing moving dots while the dots were actually stationary. ~~‘~~RW~~’~~ was unable to cancel the velocity of the eyes’ movements from the image that fell on the retina, rendering the dots motionless only when the dots moved at the same speed as the eyes.

When considering the motor plant itself, parameters often change dynamically, both internally and externally with respect to the plant. This change happens both ~~in a~~ shorter timescale (e.g. addition of masses, tool use, external forces like gravity, or damping) and ~~in a~~ longer timescale (e.g. muscular and neuronal fatigue, growing of limbs and the strengthening of muscles). To counteract such changes, it is reasonable to assume that the nervous system ~~should~~ monitor external changes and form an internal model of the plant and the environment [Shadmehr and Mussa-Ivaldi, 1994]. Numerous studies show the role of cerebellum in creating a forward model [Manto et al., 2012], getting the efference copy, understand the timing of tasks and also generate error comparing the action outcome and the estimated ~~states~~ [Wolpert et al., 1995] [Ohyama et al., 2003] [Ivry and Keele, 1989]. In ~~another~~ step tracking task performed by trained monkeys, presented evidence supporting prediction hypothesis [Fishbach et al., 2007]. Different analyses of the velocity profiles also made it evident that the corrective sub-movements were initiated when the probability distribution of the predicted end-point, from the continuously accumulated information, is statistically different from the target location. Moreover, a latency in onset of the following sub-movements proportional to the amplitude of the previous primitive and the ‘extent to go’ of the reach shows that there is a continuous feedforward controller. These studies in addition to many others, strongly point towards the existence of an internal model in addition to the feedback system, that continuously predicts the states during reaching.

The above suggests that delays in sensorimotor loops, as well as dynamic changes in the motor plant itself, necessitate the existence of internal mechanisms to handle these delays and regulate movements even before receiving sensory feedback. Hence, to accommodate for these behaviors our sensory-motor control model should include a predictive system that anticipates the consequence of ~~the~~ motor command.

3.1.2 Perception

Adapting to external perturbations and providing the appropriate control~~,~~ demands knowledge of the relevant state variables. In the context of reaching, this includes the location and velocity information of the hand. With the available noisy observations from vision and proprioception, the brain estimates the state variable with considerable ~~certainty.~~ Rushworth et al. found neural correlates of the independent contributions of both vision and proprioception in the posterior parietal cortex of monkeys. Lesions to lateral intraparietal area (LIP) and area 7 in the inferior parietal lobe (IPL) did not affect reaches where the goals were defined in the proprioceptive coordinates (in the dark), but produced mis-reaching~~,~~ when in light. In contrast, monkeys with lesions to Area 5 in the superior parietal lobule (SPL) could reach accurately in the light but not in the dark. Hence it is appropriate to suggest that the SPL may be involved in proprioceptive estimates of limb position in space, while the IPL is involved in visual estimates of limb position. An example of this in humans comes from the patient “PJ” who had an extra-axial cyst encroaching on her left SPL [Wolpert et al., 1998a].Without vision of her right arm, PJ's perception of arm position became increasingly uncertain until she reported that the arm disappeared altogether. Thus, PJ was unable to store a proprioceptively derived estimate of limb state but, presumably because of the intact IPL, could use vision to maintain a sense of the limb's position in space. Similar to Rushworth's monkeys with SPL lesion, PJ made accurate reaches when vision was present.

Furthermore, in addition to individual perception, it is interesting to study how multiple sources of information are used to have a combined perception of ~~the~~ arm states. In adaptation studies (see Figure 3.1) [van Beers et al., 2002], [Haith et al., 2008]~~–~~), after adaptation subjects estimated the hand location somewhere in between the actual location and the observed visual feedback. This was a clear indication of a weighted summation of sensory information. In fact, the model proposed by [Reuschel et al., 2010] shows that the sensory integration is close to the *maximum likelihood estimate* - the most probable prediction of the state given the prior knowledge and current observations. Further, in the process of obtaining the estimate, the brain seems to perform kinematic learning, as these two sources of sensory information are observed in different representational spaces – proprioception in angular reference frame and the visual feedback in ~~the~~ Cartesian reference frame. The estimate of the final hand state is obtained by learnt combination of the two streams of information. [Körding and Wolpert, 2004; Vaziri et al., 2006; Izawa and Shadmehr, 2008].

Overall, it evident that an effective motor control model should not only anticipate the state propagation, but gathers multiple streams of sensory information to obtain a combined perception to ~~cognize~~ the limb states, to generate errors and learn to move the limbs in a variety of situations.

3.1.3 Generalization

Another important aspect of the motor control system is its ability to generalize, or apply what has been learned in one context to another unfamiliar context. This aspect of reaching goes ~~further~~ beyond memory - not simply ~~to~~ access a built look-up table of labelled sequences of muscle activation, but ~~to~~ learn to reach under modified contexts. Examples of this generalization ~~was~~ shown in many context such as~~,~~ temporal and amplitude generalization in motor learning in [Goodbody and Wolpert, 1998]. Here upon learning to move in a novel velocity dependant force-field, subjects successfully generalized to a new state space while asked to make movements of either half the duration or twice the amplitude. Generalization has also been described between movements with movement direction [Huang and Shadmehr, 2007; Bedford, 1993], angle [Krakauer et al., 2000] and even varying movement paths [Conditt et al., 1997]. Overall generalization is a favorable trait that helps the brain to use prior knowledge to reach in a different context.

3.1.4 Adaptation

Given an internal model of the state estimate and constant feedback to correct the model and the consequence of movements, the brain ~~gives rise to~~ a plethora of interesting adaptations. There are numerous instances showing compensation for dynamic changes, including force field studies where the hand is perturbed by external forces [Shadmehr and Mussa-Ivaldi, 1994; Li et al., 2001]. ~~We~~ can also learn to make successful reaches while in a rotating room with centripetal forces [Lackner and Dizio, 1994] and ~~also~~ when other manipulations to the properties of the arm by attaching additional masses and other devices [Wang and Sainburg, 2004]). Studies have shown examples of reaching adaptation for kinematic changes, as seen in: (i) visuomotor gains [Heuer and Hegele, 2008] where the extent of reaching only altered visually~~,~~ (ii) visuomotor rotations, where the visual feedback is implicitly decorrelated and rotated with respect to the actual arm [Krakauer, 2009]. There are also more complex prism tasks[Von Helmholtz, 1867; Inoue and Kitazawa, 2018] and ~~the~~ mirror tasks [Telgen et al., 2014] where the visual feedback is altered with a complex transformation and do not represent ~~the~~ arm location explicitly. The brain is shown to adapt to making straight reaches when the hand visual feedback is provided with varying visual cues such as bars or rings indicating distance away from ~~the~~ targets [Vaidyanathan et al., 2020]. Adaptation happens even when the cursor is mapped on~~-~~to highly complex hand gestures [Danziger and Mussa-Ivaldi, 2012]. This evidence collectively showcase the extent of complexity the brain can handle when it comes to inferring limb states and performing a variety of unfamiliar tasks.

Learning to successfully reach for targets is also brought about by a understanding of decorrelation. There exist~~s~~ mechanisms that are driven by task errors and the perceived reality of limb states to achieve performance in ~~the~~ novel situation. What is ~~more~~ interesting~~,~~ ~~is~~ in all of the adaptation studies, the cortex learns to compensate for changes to the environment, sensory misalignment and the relevant task simultaneously online. It should be noted that the explicit and the implicit changes~~,~~ are inherently different in the way of adaptation since the nature of adaptation and the presence of different strategies. One involves a deliberate change of plan while the other unconsciously driven by task errors. Either way, the plethora of adaptation studies indicate that the error generation occurs at multiple levels from observation, prediction, perception, and task goals. Additionally, the brain learns to compensate on~~-~~the fly, not only when ~~the~~ perturbations are introduced, but also when removed. Ultimately, adaptation is a staggeringly essential trait that demonstrates the versatility of sensory-motor control and mechanisms for such adaptations is a crux in modelling the biological control system.

Among the various adaptation studies, two experimental paradigms (i.e., force field study and visuomotor adaptation), each reveal a unique ~~component~~ of the motor adaptive system. In ~~the~~ force field stud~~y~~, both the proprioceptive as well as the visual consequence of the hand ~~is together~~ perturbed~~, and~~ ~~n~~umerous studies have found effortless adaptation to external forces applied to the hand. Some of the most direct evidence comes from [Gandolfo et al., 2000]. They show ~~the~~ gradual recruitment of previously silent cortical neurons in M1, responding to ~~the~~ sensory error. In this experiment the estimate from vision and from proprioception coincide with each other, giving one source of error – an unambiguous difference between the observed and perceived hand location. This error ~~solely~~ drive the learning process. After adaptation, the system regains its original performance of reaching to targets by learning to compensate for the external forces. The ~~behavior of~~ adaptation to force-field provides evidence that the brain accommodates for uncertainties in ~~the~~ dynamics by correcting for errors in the task space.

The second adaptation to ~~the~~ visuomotor rotation [Shadmehr and Mussa-Ivaldi, 1994; Krakauer et al., 2000; Krakauer, 2009; Taylor and Ivry, 2011] ~~is a well studied excellent candidate for learning about the intricacies of the biological motor control.~~ This experimental setup consists of a tabletop reaching task with the goal of moving a cursor, that represents the subject's hand to targets that appear on the screen. The screen is placed directly above the subject's hand thereby blocking the direct view of the hand ~~and~~ the cursor moves in accordance to the hand's movement. The subjects reach for the targets with the help of the visual cues on the screen and their proprioceptive feedback. After a period of successful reaches, ~~the~~ visuomotor rotation ~~would be~~ introduced. In the visuomotor rotation paradigm a systematic directional bias is introduced around the hand thereby mismatching the visual feedback and the proprioceptive information of the hand location. For instance, if the subject were to move at 0°, the cursor would move 30° away with respect to the angle of reach. It is logical to infer that this ~~decorrelation~~ affects the estimate of the hand location we have in our brain since proprioception and vision indicate different estimate for the hand location. It is observed that subjects aim and reach for targets without compensating for the rotational bias initially. But after a few trials, subjects gradually adapt to this new mapping and move their actual

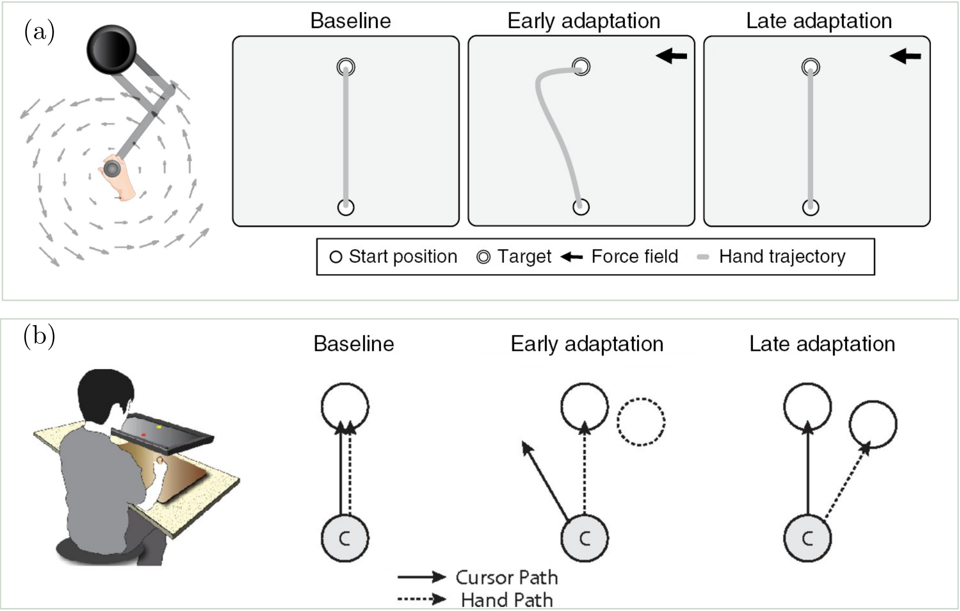

Figure 3.1: **Two adaptive behavior paradigms**: The top panel shows the set-up for ~~the~~ force field adaptation and the bottom panel describes adaptation ~~to~~ visuomotor rotation. The baselines show perfect straight reaches in both conditions. (*a*) Subjects in early adaptation have curved reaches in a counter-clockwise curlfield applied by the robotic manipulator and they learn to reach straight by the end of few trials. (*b*) Subjects learn the rotational transformation and learn to aim differently to reach for the target when the view of their hand is blocked. These two experiments~~, with different changes~~ provide ~~multiple~~ perspectives ~~of~~ primate motor control

hand in opposite direction to the transformation, adapting to the change. This learning also happens when the experiment is made more difficult by providing the cursor only during the beginning and the end of a single reaching movement.

The ~~above experimental~~ result ~~is~~ interesting for a number of reasons. Firstly, the behavior reinforces the robustness and adaptiblity of the biological motor control. In addition to previous evidence of forward models, the period during the reach where the visual feedback is unavailable, ~~the~~ task sheds light into the details of how the brain predicts the kinematics of the hand, while also dealing with unfamiliar changes in observations. In this case of sensory ~~decorrelation,~~ it seems that the brain assigns the observed error to both the visual estimate and the proprioceptive estimate, even though the actual bias is introduced only in the visual feedback. In particular, studies have shown that the brain creates a sensory illusion and perceives the hand somewhere in between the proprioceptive and visual estimate [van Beers et al., 2002]. An estimate of the hand location different from the visual and the proprioceptive observation~~,~~ hints that the brain attributes the sensory error to sensory uncertainties. The model proposed by [Haith et al., 2008] describes how the adaptation in the brain is driven by associating the visual error partially with the internal model (i.e., accounting for the unreliability of the plant) and partially to the uncertainty in the sensory information. There is rich evidence suggesting that there is continuous learning and kinematic mapping to account for these uncertainties. Schaeffer et al. show~~s~~ that the cerebellum plays a major role in learning an internal model during reaching movements. Studies in monkeys~~,~~ [Wise et al., 1998]~~,~~ also show ~~the~~ changes in M1, M2, and PMd reflect~~ing~~ the learning of visuomotor transformations. The fact that ~~the~~ rotation learning in the above experiments is subject to different forms of interference: retrograde~~,~~ anterograde through aftereffects (when the learning of new information interferes with the recall of old information), recalled with implicit contextual cues, consolidation both over time and with increased initial training, collectively exemplify the richness of this learning and the efficiency in handling uncertainties and dynamic changes. The collective evidence supports the involvement ~~and testing~~ of prediction, perception, generalization and adaptation in a single experimental paradigm of visuomotor rotation.

Overall, the sensory-motor control system is ~~well characterised~~ to efficiently control a nonlinear and unreliable plant and is robust against a variety of perturbations. As a solution to this complex problem, the system has incorporated several mechanisms that ~~is~~ constantly updated to conform with dynamic and kinematic changes in both the environment and the plant. Furthermore, the biological controller remains superior by its ~~nature~~ ~~of accommodating~~ generalization~~s and~~ adaptation depending upon context~~s~~, prior information ~~and~~ continuously learning from ~~its~~ predictive error. Modelling these features of biological learning, could not only be a ~~remarkable~~ tool ~~in~~ replicating adaptable control mechanisms, but ~~rather~~ describes the essential components that are crucial for accurately modeling the sensory-motor control system.

Chapter 4

Motor control modelling

In this chapter, we ~~delve into the realm that bridges the gap~~ between neurobehavioral observations and mathematical tools to construct ~~a comprehensive~~ model of the ~~brain, focusing specifically on~~ motor control. Building upon the foundations laid in the previous chapters, which discussed the control engineering and neuroscience background, we ~~familiarize ourselves with~~ the history of motor control models. We begin by describing the ~~hierarchies~~ of models in motor control, the current state of models and finally describe the modelling framework that we intend to leverage in building our model.

4.1 Current state of motor control models

Over the past decades, ~~the study~~ of motor control has undergone significant improvements through vast number of experiments and ~~employing~~ theoretical tools to model the pathway of voluntary movements. While ~~the~~ computational models rely on empirical data generated by observation and experiments, they capture different levels of abstraction~~s~~. Models span from describing the cellular and molecular mechanisms of neural activities, to higher level behaviors of complex movements in humans and primates. Depending upon the experimental paradigm and the objective ~~to~~ explore, ~~the~~ current computational models in motor control can be largely classified into three groups: (a) ~~C~~ost strategy models (b) ~~D~~ynamic systems models (c) ~~N~~eural population models.

~~The~~ Cost strategy models are ~~vastly~~ motivated by the theory that ~~the~~ brains should have evolved to optimize the control strategy. Given~~,~~ movement itself is an energy expensive process ranging from firing of neurons~~, or~~ ~~utilizing~~ energy to produce joint torques. The cost strategy approach seeks to find the optimal set of control inputs that minimizes an objective function while satisfying system constraints~~.~~ ~~M~~inimizing mechanical precursors of metabolic costs, such as joint torque rates, energy ([Uno et al., 1989; Alexander, 1997; Kang et al., 2005; Flash and Sejnowski, 2001]. Flash and Hogan ~~proposed~~ a model of reaching movements that incorporates the minimization of torque as a cost function. The authors in [Todorov and Jordan, 2002] showed that their model can accurately predict the trajectory of reaching movements by provide~~s~~ a theoretical framework ~~of~~ ~~'~~*minimal intervention principle*' whereby an optimal controller does not have correct for task-irrelevant deviations. ~~In~~ Prilutsky and Zatsiorsky~~,~~ ~~which~~ compared different choices of cost functions that explained the observed muscle activation pattern during reaching movements. There are several other models that look into the working of the motor control system by explaining its behavior ~~to~~ an optimizing strategy. These mathematical models typically ground their findings ~~to~~ behaviors ~~to~~ a higher level optimizing control.

~~Now~~ ~~m~~oving from ~~a~~ higher level variables such as movement cost, ~~the second level of abstraction tunes into lower~~ level movement states such as instantaneous joint torques, velocity, position etc. These ~~are dynamic systems approach that~~ try to ~~model the~~ movement system~~s~~ as a controller concerned with task objectives, state variables, and the errors that drive adaptation~~s~~. They also help in tying adaptations (or the lack of, in ~~e.g.~~ lesion studies) by attributing system level mechanisms to understand the distribution of the movement goal and contributing towards its respective anatomically distributed structures. Scott presents evidence from neurophysiological studies that bridges the ideas movement cost during reaching movements and that this optimization is based on internal models of the body and the environment. Wolpert et al. propose an internal model of a predictive Kalman filter based feedback model. Shadmehr and Mussa-Ivaldi considered the task of making reaching movements in the presence of externally imposed forces and from the simulations show elements of the adaptive process could represent dynamics of a motor task in terms of intrinsic coordinates system of the sensors and actuators. There are also probabilistic models that explain neural mechanisms that helps in estimating the sources of motor errors errors [Berniker and Kording, 2008]. Some models describe the evolution of specifics such as the timeline of learning, adaptations, and generalization. All of dynamic system models aim at replicating the behavior of the motor cortex.

?

The third approach focuses on modelling ~~the~~ motor control on a ~~more~~ cellular level. ~~They~~ enable us to connect the firing patterns of neurons or the activity of ~~the~~ brain region to provide understanding of the neural mechanisms underlying motor behavior. One of the earliest studies ~~are of~~ [Georgopoulos et al., 1989], which demonstrated that the activity of individual neurons in the motor cortex ~~and how it is~~ correlated with the direction of movement. ~~It~~ ~~hinted~~ at ~~the~~ ~~'~~population code' to predict movement direction by observing neurons from dorsal premotor cortex and the primary motor cortex. With many several experiments, we now can predict many aspects of movement tied to its neural correlates. Cellular-level models can also capture complex interactions between different brain regions, as demonstrated by Churchland et al., who proposed a model of neural population dynamics during reaching that accounted for both feedforward and feedback control. These models can also be employed to make predictions about the effects of neural manipulations, as shown by [Sadtler et al., 2014], who used optogenetic techniques to examine how learning is constrained by the neural representation of the task. In [Mante et al., 2013], they monitored an area of PFC involved in the selection and execution of saccadic eye movements.

Different approaches to modeling motor control operate at different functional levels and offer varying levels of detail. The cost strategy model, based on optimization principles, focuses on efficient control by considering the cost of movement. While it can explain higher-level constraints in motion selection, it may not fully capture the underlying low-level mechanisms of neural control. On the other hand, the dynamics systems approach can mimic individual brain structures, providing ~~a detailed~~ understanding of the neural circuitry involved. However, caution is necessary as not all system-level descriptions are biologically plausible, limiting their ability to accurately represent the motor pathways. Cellular-level approaches offer comprehensive and ~~highly~~ detailed insights into synaptic mechanisms, but can be challenging to interpret and scale up to capture higher-level behaviors. It is important to acknowledge that each approach has its strengths and aids in testing different hypotheses about the brain's function. However, it is evident that none of these approaches singlehandedly can provide a comprehensive explanation connecting overall behaviors to neural mechanisms. To achieve a more holistic understanding, a new approach is require~~s~~, one that integrates cellular activities with higher-level behaviors, organizes cortical structures, and incorporates biological plausibility to better approximate the workings of the brain.

Recently, there ha~~s~~ been a few models that tried connecting some of the aspects we discuss here. In Denève et al., they proposed a computational framework where the brain can be thought of as a hierarchical system of processing layers, each layer containing a set of computational units that learn to encode increasingly complex and abstract representations of sensory information. The authors provided evidence to support this ~~deterministic~~ framework, including simulations of artificial neural networks that exhibit similar properties to the brain, and reproduce neurophysiological data from studies of learning and adaptation in humans and animals. Although this work is an appropriate candidate ~~in this niche~~, the proposed model assumes that each layer in the hierarchy is composed of the same type of computational units ~~and~~ oversimplifie~~d~~ processing units and connectivity patterns, especially when it comes to adaptivity and learning. They also offer limited details on noise and uncertainty. While the model may be computationally effective, it could not accurately represent the actual biological processes occurring in the brain.

In contrast to the previous deterministic model, [Berniker and Penny, 2019] provided a probabilistic approach that focused on the goals and the constraints of limb movement. They proposed a network to implement an LQR control for a reaching task. This model was able to reproduce many neural firing patterns that correlate with population vectors ~~and~~ low-dimensional oscillatory activity and changes in neuron's tuning curves after force field adaptation. However, the network proposed ~~here~~ was trained assuming perfect encoding of states errors and commands and the errors from the forward model and the inverse model propagate from one system to the other. Additionally, the model ~~also~~ was motivated by an inherent stochasticity in modelling neurons from a probabilistic perspective, and hence lacked any neuronal dynamics. It is required to have the ability to capture biological details in encoding, learning and capturing neuro-scietific details when describing the behavior. While replicating a computational description of the causal relationship between neurons and the motor behavior, it is also required for us to have a biologically realistic framework.

~~Many of the~~ deterministic ~~frameworks~~ and probabilistic approaches have notable limitations that hinder their ability to ~~fully~~ capture the intricacies of the brain's biological processes. A comprehensive model should incorporate biologically plausible elements, such as ~~realistic~~ processing units, adaptivity, robustness to noise~~,~~ and uncertainty, ~~at the same time scalable~~ to encompass ~~the~~ complexity ~~of~~ the brain. Developing such a model ~~can~~ provide a ~~more~~ deeper and realistic understanding of the brain's functioning, enabling accurate representations of biological processes. In the following section we briefly discuss ~~one such~~ framework, that not only connects the lower level activity to ~~the~~ higher level behavior, but also captur~~ing~~ biological constraints.

4.2 The Neural Engineering Framework

The Neural Engineering Framework [Eliasmith and Anderson, 2003] leverages techniques from dynamical systems theory that help us model neuronal and synaptic dynamics ~~using~~ a spiking neural network. Biological constraints can be included by specifying the details of neurons, such as their dynamics, firing rate, the time constants of the neurotransmitters ~~and~~ many other features. From an engineering standpoint, NEF also allows smooth transition of these neural models to be implemented in spiking neuromorphic hardware~~s~~ making it possible to ~~also~~ have robotic implementations ~~and have~~ sophisticated controllers. In the following sections, we ~~shall~~ discuss the core principles of this framework and the tools available for us to construct ~~our~~ cortic~~la~~ structures within this framework.

4.2.1 Representation

To perceive the information about the environment, to cognize them as internal states and to utilize the states to solve problems, it is important to construct representations of the physical quantities ~~into~~ system states. ~~Here, the~~ NEF describes a mathematical framework for representation, by encoding the physical quantities into neural activities. In other words, if $x(t) \in R^q$ denotes a continuous time varying system state, it is represented in a higher dimensional vector space in the population of neuronal activity, $a(t)$ given by:

$$J_i(t) = \alpha < e_i, x(t) > +\beta_i \tag{4.1}$$
$$a_i(t) = G_i[J_i(t)] \tag{4.2}$$

where ~~s~~$_i(t)$ is the input current to the i^{th} neuron, $< .,. >$ denotes a dot product, $a_i(t)$ is the neural activity generated by the i^{th} neuron encoding the vector $x(t)$ at time t, $G_i[.]$ is the nonlinear transfer function modelling the spike-generation of a single neuron.

The function G depends upon the choice of neuron model used. A common choice is a *Leaky Integrate and Fire* or the *LIF* neuron model ([Koch and Segev, 1998]). For a given neuron, this model maintains a one dimensional voltage trace $v_i(t)$ by continuously integrating the current source J_i over time with some leakage governed by τ_{RC}. It is equivalent to a resistor-capacitor network or a low pass filter dynamics. After scaling the voltage to the interval [0,1], the governing equation ~~looks like:~~

$$\tau_{RC}\dot{v}_i(t) = -v_i(t) + J_i(t),\ 0 \leq v_i(t) < 1 \tag{4.3}$$

where $v_i(t) \rightarrow 1$ generates a spike at time t and the voltage resets to zero ~~after~~ τ_{ref} time period. Although this neuron model is an approximation to ~~the~~ neuronal dynamics, the NEF allows computation with a choice of more biologically detailed neuron models.

Along with the neuronal dynamics, we ~~also~~ ~~model the~~ post~~-~~synaptic filter that models the post synaptic current triggered by the action potential arriving at the synaptic cleft. This is modeled to be an exponentially decaying ~~P~~ost synaptic current with a time constant τ:

$$h(t) = \frac{1}{\tau}e^{-(\frac{t}{\tau})},\ t \geq 0 \tag{4.4}$$

As we represent the state variable in this fashion, we obtain a neuronal spike train that encodes the given value through time. These neuron spike trains are nonlinear encoding of vector spaces that can be linearly decoded to obtain the state representation~~s~~. We can characterize the decoding of the neural response as follows:

$$(x * h)(t) \approx \sum_{i=1}^{n}(a_i * h)(t)d_i \tag{4.5}$$

$$= \sum_{i=1}^{n}\sum_{m} h(t - t_{i,m})d_i \tag{4.6}$$

where $D = [d_1, d_2, ..., d_n]^T \in R^{n \times q}$ is the decoding matrix, used to decode the filtered version of $x(t)$ from the population activity. The $*$ operator is the convolution operator representing the synaptic filter. The collective decoders D ~~should also~~ be ~~solved~~ optimally to decode the appropriate state from the spike train, which is ~~deduced~~ in the following section.

4.2.2 Transformation

While the representation principle helps us ~~depict~~ ~~the~~ state variable into ~~the~~ neuron activities $a(x)$, we now describe how we ~~could~~ not only ~~deduce~~ the state variable but also realize new functions of the state variable. Here we leverage ~~a~~ different linear decodings to ~~compute~~ any ~~arbitrary~~ vector function~~s~~ of the given input. In order to realize a function f we can compute it in the connection between two neuronal populations. We can then approximate function f using a function decoder D^f. This decoder D^f should have the property:

$$y = f(x) \approx \hat{y} = D^f a(x) \tag{4.7}$$

~~Now~~ solving ~~for~~ a least squares problem ~~by reducing the error in~~:

$$E = \frac{1}{vol(X)} \int \int_X |f(x) - D^f(a(x) + v)|_2^2 dx, \text{ where } v\, \mathcal{N}(0, \sigma). \tag{4.8}$$

This decoding error is both function of the number of neurons in the pre population as well as the smoothness of the function calculated. With the help of the calculated decoders, arbitrary nonlinear functions can be realized in ~~the~~ neuronal connections.

4.2.3 Dynamics

We have ~~now realized a signal value in a neural form of representation~~ and devised a method to compute nonlinear functions ~~from the given input~~ as well. In this section, we add ~~the~~ temporal ~~characterisation~~ to describe dynamics. Here, we implement arbitrary dynamics of the form:

$$\dot{x} = Ax(t) + Bu(t) \tag{4.9}$$

where x is the time varying system state, $\dot{x}$ is the time derivative. Similar to how ~~we~~ the feedforward connection weights were computed, we can find a weight W^f that approximates a certain dynamical equation. For the LTI system the propogation through time happens with the help of an integrator. Similarly in ~~here,~~ the dynamics stem~~s~~ from the leaky integrator given by the canonical first order low-pass filter modelled by the synapse.

$$h(t) = \frac{1}{\tau} e^{-(\frac{t}{\tau})} = \mathcal{L}^{-1} \frac{1}{\tau s + 1} \tag{4.10}$$

In the Lapla~~s~~ian domain this can be rewritten as:

$$X(s) = \frac{1}{s}(AX(s) + BU(s)) \tag{4.11}$$

$$X(s) = \frac{1}{1 + \tau s}(A'X(s) + B'U(s)) \tag{4.12}$$

Rearranging the second equation, we get,

$$X(s) = \frac{1}{s}(\frac{A' - I}{\tau} X(s) + \frac{B'}{\tau} U(s)) \tag{4.13}$$

$$\implies A' = \tau A + I,\ B' = \tau B \tag{4.14}$$

Here we have introduced the neural dynamics matrix, A' and input matrix B' which define the dynamics of the system implemented in a recurrent neural fashion, and can be related to the standard dynamics and input matrices A and B from the state equation. By deducing the dynamics representation of a linear system, this principle allows us to map any arbitrary dynamical system that can represented in the format of eq.4.9. This gives raise to interesting circuitry realizations, such as oscillators, which are often an integral part of cortical mechanisms.

4.2.4 The Legendre Memory Unit (LMU)

An important dynamic system that we ~~shall~~ employ in our ~~mechanism~~ is ~~to~~ represent a window of signal across time. ~~LMU~~ is mathematically derived to orthogonalize the continuous-time history in a window by ~~approximating~~ it into corresponding ~~l~~egendre polynomials. This approximation of the signal allows us to learn functions from the memory of the signal in time. In this section we ~~shall~~ describe briefly the working of LMU and ~~to~~ represent it as a linear dynamical system, ~~to be able to implement~~ within the framework of NEF.

Let us consider a continuous time delay of θ seconds, expressed as:

$$y(t) = u * \delta_{-\theta}(t) = u(t-\theta),\ \theta > 0 \tag{4.15}$$

where $\delta_{-\theta}$ denotes the Dirac delta function shifted backwards in time. This function helps us in taking in a time varying scalar signal $u(t)$ and outputs a purely delayed version, $u(t-\theta)$. For this the transfer function of the system is defined as:

$$F(s) := \frac{\mathcal{L}\left(y(t)\right)}{\mathcal{L}\left(u(t)\right)} = e^{-\theta s} \tag{4.16}$$

The pure delay (see equation 4.16) has infinite order when expressed as a ratio of polynomials. To overcome this, ~~it is~~ approximate~~d~~ the irrational transfer function $e^{-\theta s}$ as a proper ratio of finite order polynomials. We do so using Padé approximates — the coefficients of a Taylor series extended to the ratio of two polynomials — expanded about $s = 0$ [Padé, 1892]~~:~~.

$$[p/q]e^{-\theta s} = \frac{\mathcal{B}_p(-\theta s)}{\mathcal{B}_q(\theta s)}, \tag{4.17}$$

$$\mathcal{B}_m(s) := \sum_{i=0}^{m} \begin{pmatrix} m \\ i \end{pmatrix} \frac{(p+q-i)!}{(p+q)!} s^i. \tag{4.18}$$

Upon choosing $0 \leq p \leq q$, we may numerically find the state-space model (A, B, C, D) that satisfies equation 4.17 given by:

$$A = \begin{pmatrix} -v_0 & -v_0 & \cdots & -v_0 \\ v_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ 0 & 0 & v_{q-1} & 0 \end{pmatrix}, \quad \begin{array}{l} B = \begin{pmatrix} v_0 & 0 & \cdots & 0 \end{pmatrix}^\top, \\ C = \begin{pmatrix} w_0 & w_1 & \cdots & w_{q-1} \end{pmatrix}, \\ D = 0, \end{array} \tag{4.19}$$

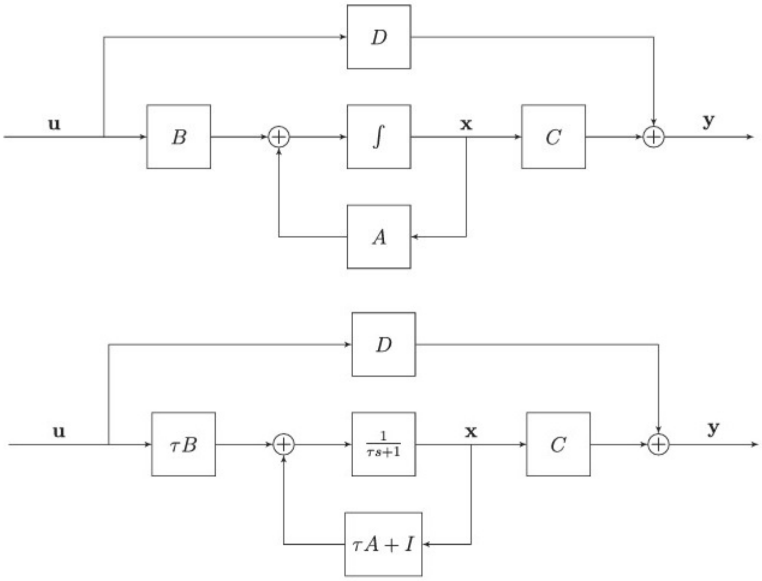

Figure 4.1: **Block diagram of an LTI system**: ~~Nef~~ dynamics Block diagram for an LTI system with the integrator replaced by a first-order low-pass filter.

This C vector helps us in decoding any instant of the signal represented in the window back in to its time dimension.

$$C_i\left(\theta'\right) = (-1)^i \sum_{l=0}^{i} \left(\begin{array}{c} i \\ l \end{array}\right) \left(\begin{array}{c} i+l \\ j \end{array}\right) \left(-\frac{\theta'}{\theta}\right)^l, 0 \leq \theta' \leq \theta \tag{4.20}$$

The basic idea ~~here~~ allows us to represent a sliding window of time θ and ~~deducing~~ different parts of the window linearly. This method of representing signal through time has a huge significance in representing and learning function of the signal represented across time. In ~~the~~ later chapters ~~of our nonlinear modelling,~~ we ~~shall~~ leverage this methodology to build ~~our~~ prediction model.

4.2.5 Learning

Synaptic plasticity is arguably one of the fundamental mechanisms in our brain that makes it possible for the cortex to ~~adapt and~~ change ~~the~~ synaptic weights to adapt over time. Many synaptic plasticity experiments result in learning rules that explain how individual connection weights change. One of the methods by which plasticity is incorporated into the framework of NEF is ~~through the help of~~ the ~~PES~~ learning rule~~. This Prescribed error sensitivity~~ [Bekolay et al., 2013]

~~or the PES learning rule~~ uses the NEF principles to help modify the decoders to learn a specific function. An error signal is generated comparing the present and the target value represented. Mathematically, this results in the learning rule:

$$\Delta d_i = -\frac{\kappa}{n} E a_i \tag{4.21}$$

where d_i are the decoders that are being learned, κ is the scalar learning rate, E is the error signal $E = \hat{x} - x$, and a is the activation of the neurons.

These core principles of NEF, ~~the~~ representation, transformation and ~~the~~ dynamics give a framework for us to represent system states and calculate different function within a spike-based network. Embedding these tools in our methodology allows us to realize neural network mechanisms in a more biologically plausible fashion and to test out the working of various cortical mechanisms. In the next section, we ~~will~~ look at ~~one of the latest models~~ of motor control that uses the NEF, ~~that~~ connects the lower level dynamics with the motor behaviors in a spiking architecture.

4.3 Recurrent Error-driven Adaptive Control Hierarchy - REACH

The Recurrent Error-driven Adaptive Control Hierarchy (REACH model) [DeWolf et al., 2016] presents an anatomically organized spiking neuron model for the motor control system using the NEF (see figure 4.2). The REACH model proposed an implementation of motor control with the help of control feedback in the operational space for moving an arm model to reach for targets. It exhibits complex high-level behaviors while also capturing lower level neuronal dynamics.

By leveraging anatomical organization of primary motor cortex (M1), premotor cortex (PMC), supplementary motor cortex (SCx), and the cerebellum, the model generates trajectories and controls that closely resemble human-like traces and velocity profiles. This is made possible through the integration of ~~a~~ spike timing-based learning rule, which facilitates two crucial aspects – cerebellar adaptation and cortical adaptation. During reaching tasks in an unknown force field (task ~~also~~ described previously in Sec 3.1.4), the REACH model showcases its cerebellar adaptation capability. Initially confronted with ~~the~~ unknown force field, the model adapts its movements by incorporating the dynamic changes introduced by the external forces. Through the ~~utilization~~ of the cerebellum, the model fine-tunes its motor commands, gradually aligning its movements to accurately navigate the force field. In addition to cerebellar adaptation, the REACH model also exhibits cortical adaptation, specifically when faced with incorrect segment lengths. In scenarios where the model starts with inaccurate segment length information, it seamlessly adapts its kinematics to ensure precise reaches within a velocity-based force field. This cortical adaptation mechanism reflects the model's capacity to rectify initial errors and optimize its motor output to achieve accurate reaching movements.

~~What's more interesting here is that this~~ model was one of the first realization with built-in biological constraints. The REACH model ~~inherently~~ uses ~~the~~ spiking neural mechanisms for ~~the~~ neural representation~~s~~ and dynamics to ~~compute~~ its control process. ~~This made it possible to group the network into well defined anatomical structures in the brain.~~ As a consequence of ~~tying the~~ bottom up and top down approaches, it was also able to realize neural firing patterns very similar to the ones observed in the monkey cortex (see Fig.??) reflecting correlations found in experimental research. ~~It has~~ population encoding of the movement magnitude, movement velocity, acceleration, rotational dynamics that ~~could~~ be seen in ~~the~~ M1. It also exhibits similar neural oscillatory patterns in low~~-~~dimensional state space observed in primates. The ability to reproduce these neural dynamics with the adaptation behaviors~~,~~ ~~pushed the model towards a more~~ holistic and biologically ~~relevant~~ approach ~~in~~ explaining the working of the primate motor control system.

~~This model made a significant improvement towards the biological plausibility and closeness to the working of motor control system. By adding neurobiological details, the REACH model encapsulated both behavioral and neural level details of the cortex. This framework employed a modified form of the adaptive nonlinear control (discussed earlier [Slotine and Li, 1987] 2.3.2) that formed the basis of adaptive control mechanism of the control system. The algorithm also contributed towards translation of biological control in robotics and hence improving the scope of biological inspiration in control as well.~~

Despite the REACH model capturing many of the adaptive control aspects of the motor control system and reproducing biological observations, it has a ~~few venues for development~~. Firstly, the model lack~~ed~~ details in the sensory cortex. ~~M~~ajority of adaptations ~~in the behaviors~~ captured by the REACH model rel~~ied~~ on ~~the~~ perfect instantaneous access to the system states~~,~~ both end effector and joint angles. With the lack of any form predictive system~~s~~, the model ~~was~~ incapable of reaching during noisy measurements or importantly in the absence of sensory observations. As we have discussed earlier (see Sec 3.1.1), prediction plays an important role in the perception and control of the limbs and actively helps in deducing plant and environment dynamics. In addition, there are no separate sensory streams of visual and proprioceptive information. The characteristic features of these sensory observations give useful information about the uncertainty and sources of error that drives adaptation at multiple levels. ~~This individual estimations~~ from various modalities are necessary to ~~describe~~ both the sensory cortex and the motor cortex, which together orchestrate various adaptations. ~~The importance can further be appreciated by pointing to the previous review of perception (Sec 3.1.2), which proves the existence of~~ separate ~~notion~~ of visual, proprioceptive and hand perceptions, which need not align and give rise to some of the important behaviors. There is a~~ necessary~~ "illusion" of ~~our~~ hand that drives the fulfilment of the task ~~in~~ hand, until any learning for decorrelation or changes in the environment happens. Behavioural adaptation in the absence of sensory perception, ~~limit~~ the abilities of the motor control model

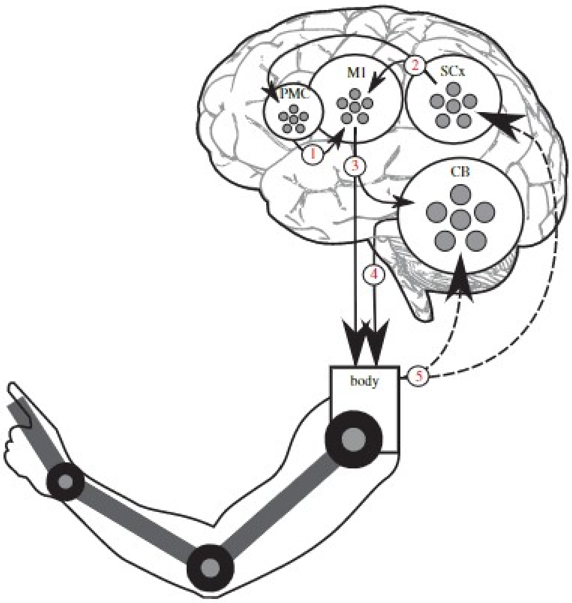

Figure 4.2: **REACH model**: An overview of the REACH model, shown controlling a three-linkarm. Numbers identify major communication pathways. Dashed lines indicate closed-loop feedback signals generated from the senses. The premotor cortex(PMC) generates a trajectory for the system to follow with a sequence of(x,y) coordinates. The primary motor cortex (M1) receives these target positions (1) from the PMC and compares them with the current system state, received from the sensory cortices (SCx), through (2). M1 combines this signal with locally calculated Jacobians to transform the desired handmovement commands into a low-level signal that is sent to the arm and cerebellum (CB) along (3). The CB projects an adaptive signal to the body along (4) that compensates for velocity and movement errors. Visual and proprioceptive feedback projects from the body along (5) to the CB and SCx.

to adapt to several complex changes which are ~~otherwise~~ observed in biology. Hence, towards the goal of constructing a detailed realization of the cortex a combined sensory-motor perception system is an indispensable step.

~~Moving forward, the~~ sensory perception systems can also be helpful well beyond adaptative strategies. While interacting with dynamically changing environments, we also learn contexts and build models of the word around us. This can only be made possible with a detailed perception systems generating errors between the anticipates and the observed. For example, error source attribution [Liepelt et al., 2008], a crucial mechanism in the brain that enables the discerning of changes and attributing the motor errors to improper prediction,

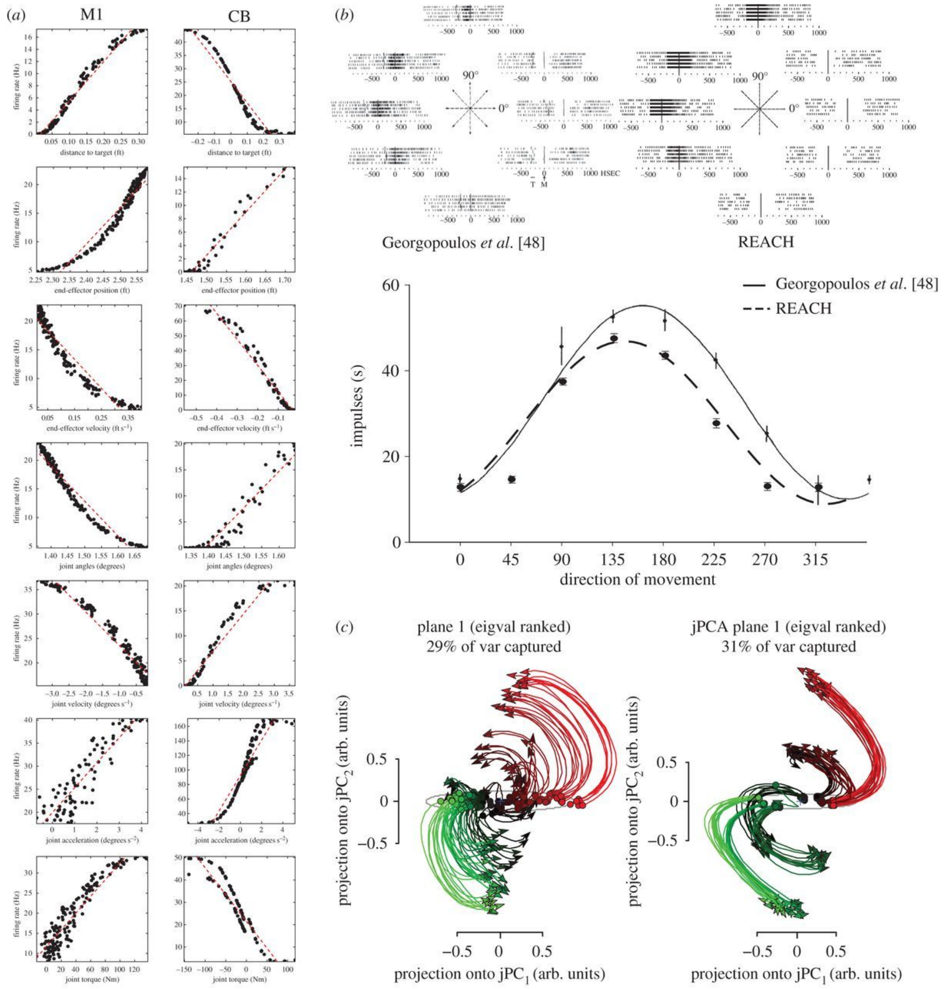

Figure 4.3: **Results of the REACH model**(a) Plots of the neurons from the primary motor cortex (M1; left column) and cerebellum (CB; right column) correlating with various high-level movement parameters. (b) Examining a neuron's activity profile in response to the arm reaching out to eight surround-centre targets over five trials. Top: original results from [48] on the left, and REACH model results on the right. Bottom: performing the same sinusoidal regression as in [48] allows a quantitative comparison. (c) jPCA analysis applied to data collected from monkeys and the model during reach trials. Results from the original analysis are on the left, and from model analysis on the right. The colour is determined by the starting position of the neural activity. Both analyses exhibit a similar rotating path through the low-dimensional state space over time.

sensory changes or from environmental factors. ~~These~~ hint at a larger system at play~~,~~ that leverages ~~these~~ detailed working of perceptual systems beyond just motor control, but to understand the physics of the environment that we are a part of. Hence, the collective shortcomings from ~~the~~ current models of motor control ~~and the unrealized essential components of the cortex~~, creates a need for a more detailed and elaborate biological modelling of the sensory-motor control system.

If the control and actuation side of the motor system helps the brain to influence the environment, it is the sensory side of the motor system that helps us observe and create a perception of the outside world we interact with. Consequently, in ~~the~~ step towards accurate modelling of the motor cortex, in the following sections, we work towards ~~the~~ sensory perception system to help us embody a more complete motor control system, instilling the ability to adapt to more complex environments, and hopefully pushing the model towards a comprehensive model of the cortex that control our limbs.

Chapter 5

Linear ~~Filter~~ Model

With reference to the experimental accounts reviewed in the previous chapter, we have identified the salient components of the motor control system: prediction, perception, and adaptation mechanisms. To summarize, the predictive system plays a crucial role in constructing a forward model of the plant dynamics while continuously estimating the necessary system states for limb control. These predictions actively guide perception by inferring system states from multiple sensory inputs and contextual information. Moreover, the predictive mechanism assists in perceptual filling-in phenomena by providing insights into sensory uncertainties and discerning system states when direct observations are unavailable. The brain detects discrepancies by comparing observations, predictions, and perceptions, facilitating adaptations that lead to improved future motor commands. In the subsequent sections, we ~~will~~ delve into the construction of a preliminary linear ~~filter~~ model that incorporates these components to encompass the system ~~of interest~~.

Based on the analysis of biological behaviors, let us distill the necessary functional elements of the prediction system. The prediction mechanism has a running recurrent estimate that forms the basis of the forward model. To construct an effective forward prediction model, it is essential to incorporate the concept of an efference copy of the control signal. This copy of the input command is used to propagate the system states, enabling the formation of an internal prediction of the expected sensory outcomes resulting from the applied motor commands. Given the dynamic nature and noise present in sensory information, the prediction mechanism should effectively handle observation noise. Additionally, the brain ~~optimally~~ estimates system states by weighing the certainty of incoming sensory observations. In the presence of increased noise in the observations, greater trust is placed in estimates, while more accurate observations allow for adjustments of the prior beliefs. Furthermore, the intended motor control model must also account for dynamic changes in limbs that occur over multiple timescales, including variations in system parameters (e.g., mass, length, stiffness) and kinematic or dynamic alterations in the target dynamics. It is important to note that arm dynamics are often nonlinear, underscoring the complexity ~~and~~ inherent ~~nature of~~ the prediction system. ~~In total these descriptions, formulate the requirements of the prediction system.~~

5.1 Linear prediction model

Let us draw upon the engineering methodologies established earlier, to build a biologically relevant prediction mechanism. Among the various systems explored, the Kalman filter emerges as a compelling starting point for an estimation model due to its alignment with many of our prediction model requirements. The Kalman filter offers a reliable framework for real-time estimation and control, operating recursively to refine predictions with each new measurement. Notably, the filter's ~~computational efficiency~~ arises from its optimal estimation, ~~with~~ th~~e knowledge of~~ the noise covariance of input measurements and the dynamics. Moreover, the robustness against noise better equips the filter to handle the challenges of an online motor control system to deal with noisy sensory data, providing an accurate estimate~~s~~ of the system's true state. Unlike the constant calculation of system gradients in an Extended Kalman filter and the stochastic nature of Particle filters, the ~~simple working~~ and ~~the~~ deterministic nature of the ~~basic~~ Kalman filter further reinforces its suitability, compared to its computationally heavy and biologically implausible counterparts. Given these merits, the Kalman filter serves as an ideal real-time estimation system and a promising starting point for our prediction model.

Indeed, the Kalman filter is a viable candidate, but it is important to consider ~~the~~ limitations and its impact on the prerequisites of building a biologically realistic sensory-motor prediction model. One limitation is the filter's reliance on assumptions of linearity for system state propagation. This linearity constraint results in inaccurate predictions when dealing with real-world nonlinear interactions in hand control dynamics, kinematics, and sensory transformations. Additionally, the Kalman filter assumes Gaussian distributions for the measurement and process noise, while noise sources in practice may exhibit non-Gaussian characteristics, such as heavy tails or skewness [Victor and Purpura, 1996; Johnson, 2001]. These violations of the Gaussian noise assumption can significantly affect the filter's performance and result in less accurate state estimates. Furthermore, the Kalman filter requires precise knowledge of system dynamics and the influence of control inputs on the system, which may not be fully known in the context of motor control. Moreover, the motor control plant is also known to be susceptible to change over time. In light of these inherent shortcomings, along with the overhead of tailoring the mechanism to a biologically realistic framework, using a Kalman filter model presents significant challenges.

To strike a balance between capturing complexity and ease of implementation, we consider a few alterations and assumptions ~~towards~~ building an initial version of the model. In an effort to match the linearity assumption of the filter, our preliminary model will approximate the nonlinear arm dynamics of the plant with a unit mass system moving on a frictionless surface perpendicular to the plane of gravity. This simplification linearizes the target plant, making the system suitable for a Kalman filter framework ~~and instilling~~ the relevant motor control system states such as position, velocity, and control forces. In the future, this system will also be appropriate for capturing the estimation of a ~~’~~cursor’ or a visual marker dynamics representing the arm’s end-effector, which is inherently linear. To address the complexity of noise characteristics, we assume an independent Gaussian distribution. While this simplification allows for ease of estimation and hypothesis testing, it is important to acknowledge that real-world noise encompasses various complexities across multiple levels, including system dynamics perturbations, visual and proprioceptive sensory noise~~s~~, and transmission noise~~s~~. Additionally, to account for instances when observations are unavailable, such as when visual cues disappear or when the eyes are closed, we prevent system correction driven by errors by inhibiting the error population. This inhibition ensures that the estimation system remains robust during sensory unavailability. Furthermore, for this initial implementation, we relax the constraint of dynamically changing systems and instead work with a known time-invariant system. These assumptions and alterations allow us to develop an ~~iteration~~ of a Kalman filter-based estimation system ~~with simplicity and clarity,~~ provid~~ing~~ an initial step towards a ~~practical real-time~~ implementation.

Despite the ~~relaxations~~ mentioned, ~~the~~ Kalman filter estimation serves as a valuable starting point for the forward prediction model and is ~~particularly~~ relevant to our ~~descriptions.~~ The objective here is to establish a basic filter-controller mechanism that accurately deduces system states from noisy observations in a biologically ~~realistic~~ manner, and drives the system to the desired target. To achieve biological ~~realism,~~ we ~~will~~ implement the algorithm using the NEF framework ~~using~~ spiking neurons. We simulate the system physics numerically and handle the representation and prediction in spiking neural ensembles. We ~~shall~~ evaluate the prediction system’s performance in the absence of observations, adding noise, changing the noise covariance ~~and also~~ implementing it in a motor control model to reproduce biological behaviors. It is important to note that many of these assumptions will be ~~refined,~~ and a more realistic estimation framework ~~will be~~ developed in the later parts of th~~e~~ thesis. For now, the realized linear prediction system ~~can act as a middle ground~~ for testing ~~the~~ initial performance while also incorporating a neuro~~-~~biologically ~~realistic~~ framework for realtime feedback control.

5.2 Neural implementation ~~of~~ Kalman Filter

~~Now, consider~~ a unit mass system moving on a frictionless surface~~,~~ as the system to be controlled in our neural Kalman filter implementation (see Fig.5.1). The system’s dynamics ~~is~~ described by linear state space equation~~ of~~:

$$\dot{x}(t) = A\,x(t) + B\,u(t) + Gw(t) \tag{5.1}$$

$$z(t) = Cx(t) + v(t) \tag{5.2}$$

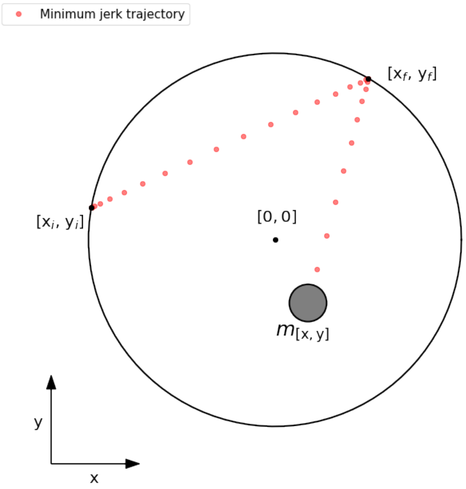

Figure 5.1: **System environment**: Control of a unit mass system to ~~reach for~~ different targets on a unit circle. The grey circle represents the mass and the red dots represents the minimum jerk trajectory from one target to another, along which the mass is made to move.

where $x = \left[\mathrm{x}, \mathrm{y}, \dot{\mathrm{x}}, \dot{\mathrm{y}}\right]^T$ is the state of the system consisting of the Cartesian position and velocity, $u = \left[\mathrm{u_x}, \mathrm{u_y}\right]^T$ are the forces acting in the x and y directions and $z = \left[\mathrm{z_1}, \mathrm{z_2}, \mathrm{z_3}, \mathrm{z_4}\right]^T$ is the corresponding observation vector. The A, B, G, C matrices are the state transition, control, dynamic noise and observation matrices. $w(t)$ is the process noise, sampled from a Gaussian, zero-mean and of covariance Q_n and $v(t)$ is the measurement noise, sampled from a Gaussian, zero-mean and of covariance R_n. Furthermore, $w(t)$, $v(t)$ and $x(0)$ are chosen to be mutually uncorrelated. The A, B, G, C matrices are chosen appropriately to describe a unit mass dynamics observing all the four system states.

Given the ground state $x = \left[\mathrm{x}, \mathrm{y}, \dot{\mathrm{x}}, \dot{\mathrm{y}}\right]^T$, the objective of the controller is to move the mass~~,~~ to randomly generated targets on a circle~~,~~ along the shortest minimum jerk trajectory to the target. The implemented controller is a proportional derivative (PD) control given by:

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = K_p \left(\begin{bmatrix} x_r \\ y_r \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} \right) + K_d \left(\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \end{bmatrix} - \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \right) \tag{5.3}$$

where the x_r , y_r and $\dot{x}_r$, $\dot{y}_r$ are desired minimum jerk positions and velocities. The trajectory and the control gains K_p and K_d are chosen so as to drive the mass to the target within 1 second. When the mass hits a position threshold and a velocity threshold, a trial is considered complete and a new target is generated.

The observed state z from the simulation is represented in an ensemble of neurons. The estimation mechanism does not have access to the ground truth, but rather the noisy observations only. The goal of the prediction system is to provide a state estimate $\hat{x}$, given the noisy observation z, and the forces u applied on the mass to move in the desired trajectory to the target. The estimate process noise covariance Q_{est} and the estimate measurement noise covariance R_{est} are variables that represent the uncertainties, and can be varied for different trials to evaluate the performance of the filter. It is to be noted that the ~~earlier~~ Q_n and R_n are the actual noise covariances that gave r~~a~~ise to the simulation measurements, whereas the Q_{est} and R_{est} are ~~perceived~~ covariances that are used for our estimation. The covariances from the simulation and the estimation need not necessarily match ~~with~~ each other, since the estimation network does not have ~~the~~ ground truth knowledge of actual covariance even from a biological perspective. ~~Especially it~~ is convenient to have ~~these~~ as separate variables as we intend to vary the Q_{est} and R_{est} to evaluate their influence in the estimation process.

Recall the previously discussed Kalman filter estimation algorithm. The propagation of the ~~estimate~~ covariance is with the help of the continuous recurrent Kalman filter equation given by:

$$\dot{P}(t) = AP(t) + P(t)A^T + GQ_{est}G^T - P(t)C^T R_{est}^{-1} CP(t)^T \tag{5.4}$$

The covariance of the current ~~belief or the~~ estimate is given by P. From the eq.5.4, it is clear that both the process and measurement noise covariance affect the estimate covariance propagation. Note that ~~the choice of~~ Q_{est} and R_{est} ~~if~~ maintained at a constant value, ~~the~~ P either increases or decreases accordingly but settles at a steady state value eventually. The rate of change of this estimate covariance is also a function of the state transition matrix A. With the help of this estimate covariance matrix, we can calculate the filter gain K using:

$$K(t) = P(t)C^T R_{est}^{-1} \tag{5.5}$$

$$\dot{\hat{x}}(t) = (A\hat{x}(t) + Bu(t)) + K(t)(z(t) - \hat{x}(t)). \tag{5.6}$$

In the implementation, a separate ensemble calculates the Kalman gain (K) with the help of the estimate covariance matrix (P). The network structure of the filter is represented in ~~the~~ Figure 5.2. The neuron populations that represent~~s~~ the state estimates of position and velocity recurrently predict~~s~~ the states, with the knowledge of the state transition matrix A and the efference copy of the command $u(t)$, ~~that~~ is sent to the plant. ~~This together forms~~ the prediction part of the estimation system that propagates the system states. Given ~~the~~ eq. 5.4 and eq. 5.5, we ~~can~~ have a rough intuition of how the estimate covariance and the Kalman gain propagate~~s~~. The estimate covariance P acts as a weight that dictates how much of the new estimate is reliant on the observation ~~over~~ the estimates provided by the prediction system ~~by propagating~~ the previous estimate using the system model. When the process covariance

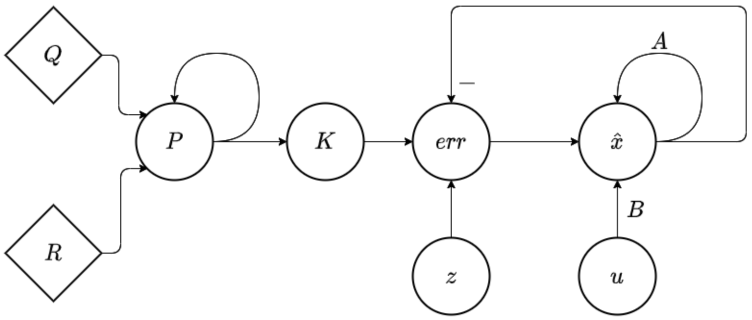

Figure 5.2: **Spiking neuron model of the Kalman Filter**: The Q and P nodes represent the estimate dynamics uncertainty Q_{est} and the estimate of measurement uncertainty R_{est}. The ensemble P represents the running estimate covariance of the states and K represents the Kalman gain. The $\hat{x}$ is the estimate of the system given the calculated control u and measurement z from the simulation. The circles represents neuron ensembles while diamonds represent nodes.

Q_{est} increases (or the measurement noise covariance R_{est} decreases), P and K increase~~s~~. This results in adjustment of the prediction by correcting with respect to the newly obtained observation. Conversely, when the process noise Q_{est} decreases (or the measurement noise R_{est} increases), ~~in turn reduces~~ P and ~~hence~~ the Kalman gain. This way, the new estimate is driven ~~more~~ closer to the prediction than ~~corrective against prediction errors from~~ observation.

The measurements $z(t)$ ~~is~~ compared against the prediction $\hat{x}$, generating the error signal required for correction that is calculated in the *err* population. This correction is now scaled with respect to the calculated Kalman gain.

5.3 Linear Filter Results

The Figure 5.3 shows the control ~~of the~~ mass on a 2D frictionless plane, with the numerical implementation on the left and the spiking neural implementation on the right. The dynamics simulation that provides the observation for both ~~the~~ implementation is run python. The simulation takes in the control forces as ~~the~~ input and generates the position and velocity observations of the mass, with the added noise (according to Q_n and R_n) and the desired minimum jerk trajectory to follow. In this section, since we purely evaluate the the performance of the estimation system and its accuracy compared to the ground truth, the system dynamics ~~is~~ controlled by the ground truth, as opposed to the estimate obtained from the Kalman filter. So the system can be viewed as ~~two parallel~~ independent filter and controller system, and the imperfections in the estimate do not affect the system dynamics.

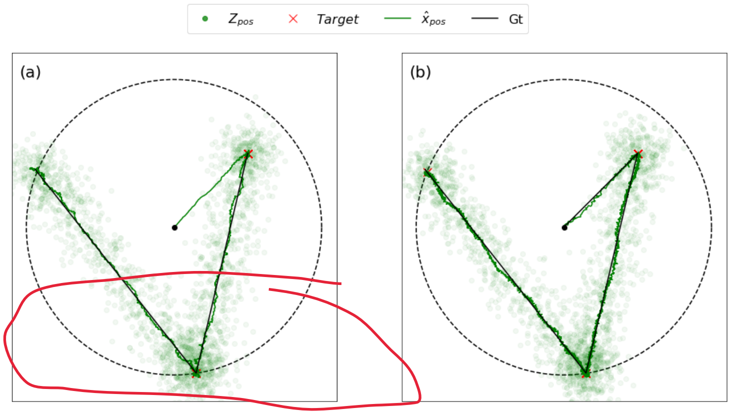

Figure 5.3: **Results of Numerical simulation vs Neural Implementation**: (a) Numerical implementation (b)Neural Implementation. The noisy observations are depicted by green dots. The Targets are represented by 'red x' and the actual position or the ground truth is depicted in solid black.

The left panel in Fig.5.3(a) shows empirically run Kalman filter model and panel on the right 5.3(b) shows the neural implementation. The measurement used for estimation is shown in green dots. The state estimate is initialized at origin, where $\hat{x} = [\hat{\mathrm{x}}, \hat{\mathrm{y}}, \hat{\dot{\mathrm{x}}}, \hat{\dot{\mathrm{y}}}]^T = [0, 0, 0, 0]^T$, whereas the actual mass is randomly placed away from the origin. From the plots, it is evident that the model's estimate $\hat{x}$ (in solid green) in either simulation~~,~~ falls close to the ground truth (in black). Upon observing the states of the mass, the filter quickly recovers from the offset and corrects the estimate within a few steps. This correction for error can be more clearly seen from the Fig.5.4 and the error plot in Fig.5.7. The estimates correspond~~s~~ to an $R_{est} = 0.01$ and $Q_{est} = 10$. The predictive system, despite getting a comparatively noisy observation, is able to predict the states of the system accurately. Comparing the states from both the panels, in ~~fi~~gure 5.4 and the error plots from ~~5.4,~~ the performance of the neural implementation is comparable with the empirical simulation of the Kalman filter.

There are a few differences between the results of the two implementations as seen in Fig 5.4. Although the dynamics of the state propagation is similar, it is important to note that there is additional noise in the neural implementation compared to the numerical implementation. This is due to the fact that the estimation system is running in neurons, which has an inherently noisy representation. ~~In fact this~~ comes from ~~the~~ biological ~~realism~~ of the working of the mechanism and further helps us induce the robustness of the model against measurement noise. The slight difference in the beginning of the trial (about 0.2 sec), corresponds to the transient period in the neural simulation, and hence the control is turned on only after the transient period is over. Within this period the initialized observation catches up with the observation ~~comparatively sooner~~.

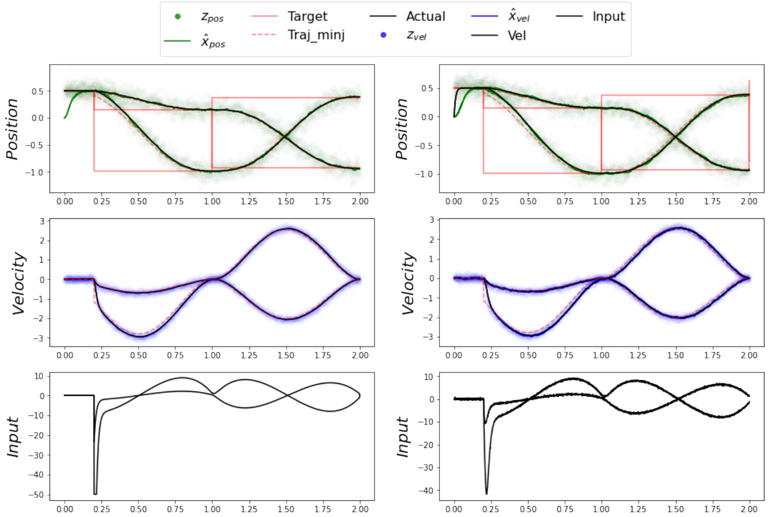

Figure 5.4: **State propagation and estimation of the filter**: The left column is the numerical simulation and the right column is the neural simulation. The figure shows the ground truth and the estimates across time. The control input is given in the bottom panel

5.3.1 Performance ~~across~~ varying measurement uncertaint~~ies~~

One ~~of the other~~ requirement~~s~~ of the model is to ~~optimally~~ combine ~~the~~ predictions with ~~the~~ measurements. The system's estimate should consider the uncertainty in the prediction and the measurements and the Kalman gain is ~~weighed appropriately~~ based on the~~se~~ noise covariances. ~~We~~ test the system for this ~~feature~~ by varying the process noise covariance~~s~~, and the measurement noise covariance~~s~~ across different runs. The resulting position estimates are shown in the ~~fi~~g.5.5 and the Root Mean Square Error(RMSE) is calculated for the noise covariance sweep, ~~given~~ in Fig.5.6.

The top panel (both Fig.5.5 and Fig.5.6), shows how the estimate changes with ~~the~~ increase in Q_{est} ~~estimate~~ dynamics covariance with a constant measurement covariance, and the bottom panels, shows how the model' estimate changes when the R_{est} measurement covariance is increased but the ~~estimate~~ dynamic covariance is held constant. The increase of Q_{est} is tested for the range of [0-1000] while the R_{est} is varied in the range of [0-1], since Q_{est} affects the derivative and the noise is scaled by a factor of dt. The RMSE is calculated for both the position and velocity after the transient period of 200ms to prevent the initial dynamics of the neurons and initiation error affecting the RMSE. Hence the RMSE error reported here is a quntitative ~~metric~~ of the steady state error ~~given the same meaurements.~~

The 2D position plot in fig.5.5, with the increase in Q_{est} in the top panel,

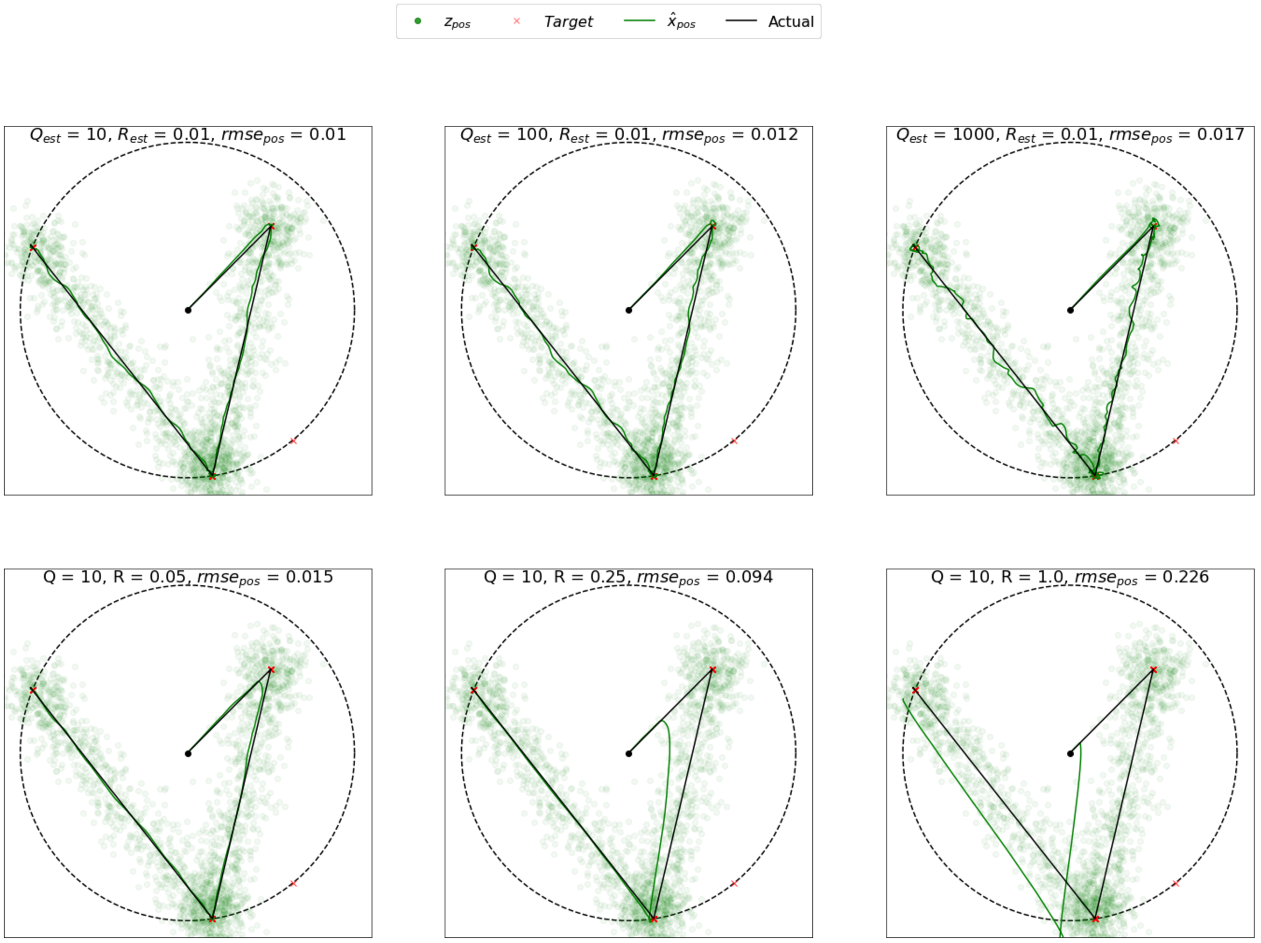

Figure 5.5: **Performance of the neural implementation of Kalman filter**: The top panel shows the performance of ~~them~~ model with a constant measurement noise (R_{est} = 0.01) while changing Q_{est} and the bottom panel has a constant dynamic noise (Q_{est}=10) while varying R_{est}. The state estimation is smoother and closer to ground truth when both the uncertainties are low. The filter relies on measurement more with lower R_{est} and ~~sticks~~ more ~~with~~ the prediction when Q_{est} is low.

the estimate weighs the measurement more. This is also evident from the increasing trend in RMSE error in 5.6. The ~~"~~unit mass" nature of the dynamics is lost and the estimate looks increasingly noisy. In the bottom panel, the R_{est} is increased, indicating high measurement uncertainty. Thus the calculated gains drive~~s~~ the estimate to ~~believe~~ the measurements ~~less~~ and rely more on the prediction. Especially~~,~~ with the intialization of the position estimate at zero and the actual mass farther from the origin, supports this inference further. In the third instance (c) shown in the bottom panel in ~~fi~~g.5.5, has an increased reliance on the incorrect estimate intialization, indicating a stark deviation from the ground truth. It is also interesting to point that since the velocity estimate is closer to the ground truth ~~and hence~~ the estimate heading direction is in the same direction ~~of~~ ground truth. Additionally, the error gradually diminishes pushing the estimate towards the ground truth eventually.

The trends ~~of~~ the performance of the neural implementation of ~~linear~~ filter can be seen in ~~the~~ ~~figure~~ 5.6. With an increase in either ~~of the~~ noise covariance, Q_{est} and R_{est}, the RMSE error increase exponentially. There is also a considerable error difference in position compared with the RMSE error in velocity. This is caused by three different factors. Firstly, the initialization of the position farther from the ground truth, resulting in slow convergence of the estimate in position ~~opposed~~ to ~~the~~ velocity. Secondly, the R_n is a chosen constant of 0.01 for both position and velocity, where the magnitude of position and velocity is different (see ~~fi~~g.5.4./ Finally, the position vector is more sensitive to errors, since the incorrect estimate in velocity also adds into the position estimate by ~~the~~ virtue of the system dynamics.

Collectively, these results show that we not only have a neural implementation of the Kalman filter close to the working of a numerical implementation, but also the biologically plausible filter system works in accordance with our requirements. The filter can effectively handle measurement noise~~s~~, and optimally combines the prediction and the measurements based on the proper calculation of gains from the estimate process and measurement noise covariances. ¶ From our previous discussion of the features of our prediction and perception system, we have to implement ~~the~~ additional features namely:

- State estimation under absence of measurements
- ~~The~~ online estimates driving the control in place of the ground truth completing the feedback loop
- The prediction system actively driving the perception system, and hence facilitating adaptation

To add on the above features ~~of~~ our prediction-perception~~s~~ system, we ~~shall~~ implementation to the first iteration of the sensory-motor control model. This way, we can test out the prediction system as well as incorporate the adaptation~~s~~ mechanisms thereby building ~~details~~ towards ~~the~~ ~~biological perspective as well.~~

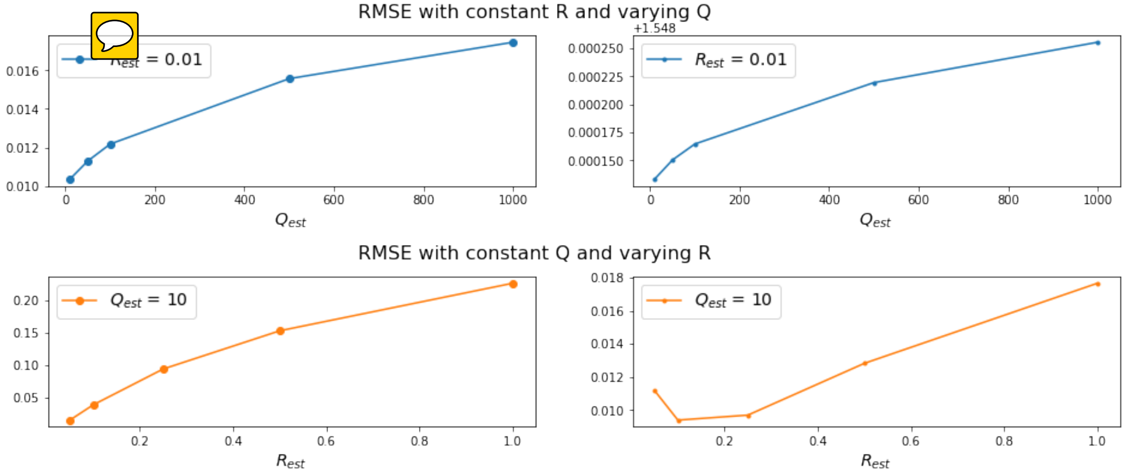

Figure 5.6: **RMSE error for different noise covariances**: The top panel shows the error change with the increase in Q_{est} with a constant measurement covariance $R_{est} = 0.01$ ~~, and the~~ bottom panel, shows error increase with increase in R_{est} and a conostant $Q_{est} = 10$.

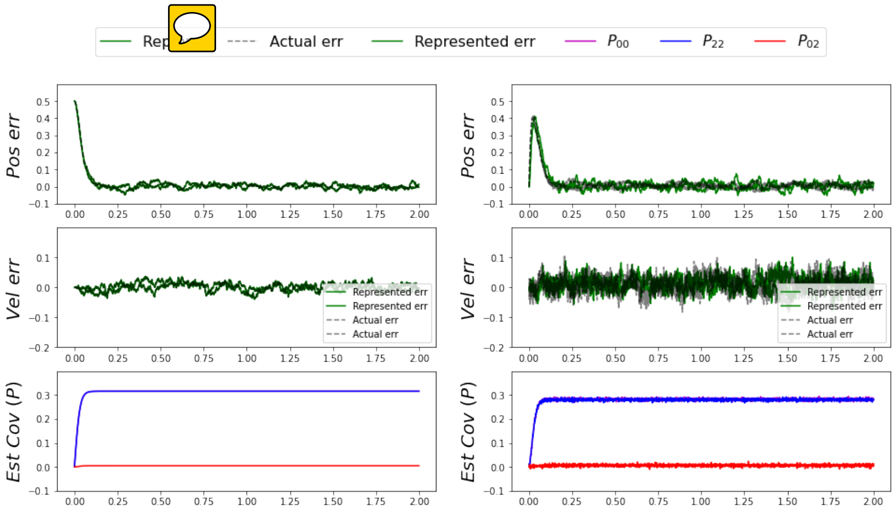

Figure 5.7: **Estimate errors and covariances**: The left column shows the error in the numerical implementation while the right column shows the error in the neural implementation. The top two panels shows the position and velocity errors in the estimate while the third shows the estimate covariance. The two implementations are comparable in their performances.

5.4 Sensory-motor control ~~model with linear filters~~

The sensory-motor control system deals with two streams of information to perceive the hand or the end-effector states ~~-~~ vision and proprioception. These two sensory inputs help the system estimate the states required for controlling the arm to accomplish tasks in the operational space. In the context of reaching, visual observation helps deduce target information, hand location, and velocity necessary for planning a reach. Propriocepti~~ve~~ joint angles and rotational velocities allow inference of joint locations and the transformation of end-effector forces to arm torques to follow a specified trajectory. The optimal combination of visual and proprioceptive estimates enables the brain to perceive the hand's location, which ultimately control~~s~~ arm reaching. To realize the entire system and incorporate biological ~~details~~ ~~of~~ prediction, perception, and adaptation, evaluating the model's behavioral implications becomes crucial. Hence, to explore the model's behavioral capability, we ~~will~~ test its performance ~~in one of~~ the canonical behavior~~s~~ of visuomotor rotation (VMR).

5.4.1 The Visuomotor Rotation paradigm

As discussed earlier (in ~~s~~ec3.1.4), the task of ~~V~~isuomotor rotation (or VMR)~~,~~ is a familiar rotational perturbation experiment where a transformation is introduced in the visual feedback of the hand location. Generally, this is either induced by having the subject wear a distortion prism while reaching for different targets, or by blocking the view of the hand and providing ~~a~~ altered cursor feedback representing the hand (See Fig.5.8). Initially during the baseline reaches, there is no distortion and hence the visual and proprioceptive feedback~~s~~ are in agreement with each other ~~and~~ ~~h~~ence the reaches for the targets are ~~a~~typical and head~~ed~~ straight towards the target. ~~During~~ the ~~initial~~ exposure, the visual consequence of the hand movement is rotated about an origin in the task space, leaving the proprioceptive feedback unperturbed. As soon the subject is exposed to this visual perturbation, there arises an incongruency between the visual and proprioceptive inference of the hand location. ~~The subjects~~ while initially aiming at an offset creat~~ing~~ directional error, within a few trials compensate for this rotational transformation by changing their motor plan. By the end of the exposure ~~the~~ subjects direct their hand to a ~~relatively~~ different angle from the origin, thereby moving the cursor straight towards the target. What's more interesting is that when this incongruency is removed, to make the feedback ~~to~~ align with the actual arm location, ~~the~~ subjects aim at the compensated angle. Now this learn~~t~~ compensation has to be unlearned in the washout trials, to return to the original perception and task performance.

This experimental paradigm of VMR particularly aligns with our goal of capturing the intricacies of the multiple perception and adaptive mechanisms of the sensory-motor control system. Consider the ~~outline~~ architecture of the sensory motor control system is described in the figure 5.9. Before the introduction of

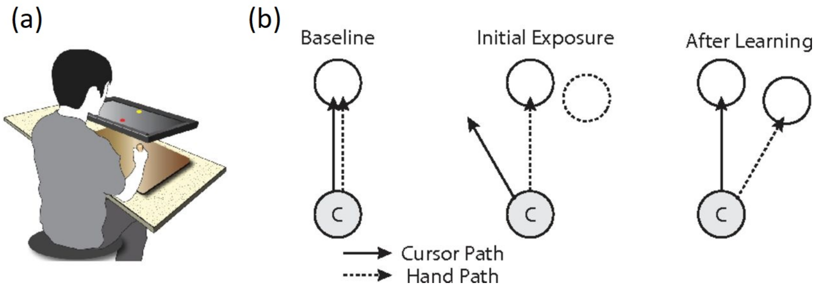

Figure 5.8: **Visuomotor rotation experiment**: (a) Experimental setup where the visual feedback is provided on the screen, blocking the actual view of the hand (b) The baseline, ~~I~~nitial exposure and ~~A~~fter learning washout trials demonstrating ~~the~~ adaptation to the visuomotor rotation of the visual feedback. The solid line shows the path of the cursor while the dashed line represents the path of the hand

the rotation, the proprioceptive estimate, the visual estimate and the hand estimate are all aligned. ~~Now w~~hen the ~~collective~~ hand estimate drives the system, ~~this~~ completes the feedback loop driving the arm to move to the desired location. This system on its own help us delineate the two perceptive system and a combined hand estimate, encapsulating the closed~~-~~loop sensory perception and control system. Once the transformation is introduced, the estimates become~~s~~ ~~askew~~ triggering mechanisms at multiple levels. Firstly, the visual predictive system now need to predict differently. For the same forces applied to the arm, the visual feedback behaves different from ~~the~~ previous experience and hence the predictive system~~s~~ should anticipate this new change. Secondly, in many of the VMR experiments the feedback of cursor is made invisible from the start of the reach and is provided at the end of the reach. The system~~s~~ observe that even when the supposed hand location is on the target, the cursor is misaligned from the target. To perform the task successfully, the control system needs to learn~~s~~ the transformation of the movement vector~~,~~ such that the cursor lands on the target (and not the hand)~~, accomplishing the task in hand.~~ Furthermore, in the control of the limb in real world, is a Jacobian that maps the joint angles ~~in~~ the proprioceptive estimate to provide the hand estimate~~s~~. Since there is a misalignment, we also need to learn this kinematic transformation to perceive our hand better. These various adaptations driven by different sources of errors, change in perception~~s,~~ modification of prediction ~~collectively~~ makes the VMR experiment a wonderful candidate to ~~model the~~ sensory motor control.

While the behavior itself is rich and informative about ~~the~~ adaptation~~s~~ in ~~of~~ sensory-motor control system, it also contributes toward~~s~~ ~~the building~~ complexity of the the model. ~~Hence this points us towards~~ some ~~of the~~ assumptions we need to make during the building of the first iteration. In this preliminary model, we ~~resort to~~ remove the complexity of working in different coordinate~~s~~ for the multiple modalities (joint and cartesian coordinates) and hence we have approximated the arm dynamics to a unit mass linear subsystem. This relaxation is also consistent with our previous filter implementation allowing us to incorporate the Kalman filter model for the prediction of the dynamics. We ~~shall~~ also remove the adaptative prediction that learns to dynamically predict differently through the course of inaccurate observations. The prediction systems ~~s~~ will have a constant state transition dynamics. To compensate for this, and to capture the resulting behavior, we ~~shall~~ have an increased reliance on the observations, which forces the estimate to be ~~more~~ closer to the observations the predictions. It is necessary to emphas~~is~~, that these ~~relaxations~~ are for the first iteration of the model, and we ~~shall~~ remove ~~these approximations~~ to realize a more accurate model in ~~the~~ subsequent chapter. ~~Now~~ we ~~shall~~ structure the sensory motor control system to implement a neural realization of the prediction, perception and adaptation to exhibit the visuomotor rotation behavior and to have a biologically ~~realistic~~ working ~~of the~~ model.

5.4.2 Experimental setup

The environment consists of a unit mass system that moves on a frictionless 2D surface, along the transverse plane and perpendicular to gravity. The mass is made to move along the surface by applying forces to reach for the targets. The simulation is run with a dt of 1ms. Proprioceptive observation is measured unperturbed ~~wild~~ visual observations or rotated whenever necessary. The rotation is made at a angle of 45 degrees in a counter clockwise direction with respect to the origin. The targets appear at 8 directions ~~along~~ a unit circle in a pseudorandom order and the hand is expected to make center~~-~~out reaches to these targets. A single trial is defined as the movement of the cursor, that represents the visual feedback, from the origin to the target. The reach towards the origin does not count towards the trial for the behavior. A minimum jerk trajectory is generated so that mass lands on the target within 0.5sec. The trial ends when either the cursor lands within a position and velocity threshold or the reach time exceeds ~~the~~ maximum time of 1 s~~ec~~ (reach time 0.5sec+ wait time 0.5sec). The visual feedback is provided only at the beginning and the end of the reach and is made invisible during the majority ~~duration~~ of the reach.

There ~~there~~ are three stages ~~d~~o the experiment. The first 16 trials correspond~~s~~ to the baseline reach. During this time the visual feedback coincides with the ground truth ~~head~~ location and no rotation is introduced. After the baseline stage~~, in~~ the initial exposure ~~the~~ visoumotor rotation is introduced with a sequence of 80 trials. Following this, during the washout stage, the visual perturbation is removed constituting another batch of 80 trials. The heading direction for each of the trial is calculated when the cursor feedback appears at the end of the reach. The time course of the change of heading direction for each of the reach is monitored across the stages to study the extent of adaptation.

The visual and proprioceptive observations are provided with the x-y position and velocity of the mass with noise~~s~~ as described by the Q_n and R_n for each modality. This measurement is sent to the spiking neuron model the sensory motor control system, which give the final movement vector that is fed back into the simulation.

5.4.3 Model description and ~~working~~

To build an initial architecture of the sensory-motor control system, we implement individual prediction, perception and adaptation systems. The pictorial representation of the model is given in fig.5.9. To incorporate ~~the~~ visual and proprioceptive estimation~~s~~, the model has two perceptual systems - vision and proprioception. The plant provides ~~the~~ two sensory observations y_{prop} and y_{vision} and we implement two separate neural Kalman filters for estimating the states through each of the individual modalities. The filters generate~~s~~ prediction based on the built-in plant dynamics and the efference copy of the forces applied on the system u, and update~~s~~ the prediction based on ~~the~~ noisy measurements. The estimate covariance and the kalman gains are calculated based on the Q_{est} and R_{est} for each sensory ~~estimation~~. ~~The working of each~~ of the neural spiking ~~k~~alman filter is identical to ~~the working described before~~.

The estimates ~~thus~~ obtained from the two modalities are combined together to obtain ~~a perception~~ of ~~the~~ hand. This combination is performed by weighing the sensory estimates with the ~~relative~~ ~~belief~~ in the vision ~~vs the~~ proprioception. The resulting combined estimate of the hand is given to the controller. The controller uses a proportional derivative control to generate the movement vector that drives the arm along a minimum jerk trajectory to the target. As dicussed earier, in this simplified preliminary model of the sensory-motor control system the only adaptation is the adaptation of the movement vector to follow a trajectory to compensate for the introduced rotation. This is learned by the ~~simple~~ PES learning rule, with ~~the~~ hand estimates as the input feature and cursor incongruency error driving the learning. With the combined prediction, perception, and adaptation mechanisms realized in a spiking neural framework, this model describes our first iteration ~~for the~~ sensory~~-~~motor control system.

5.4.4 Results

The system is simulated in the ~~given environment~~ and made to reach for the baseline, exposure and washout stages so as to monitor ~~in~~ heading direction and its change with the introduction of rotation.

Single reach results

Let us consider the working of the sensory~~-~~motor control model with respect to a single trial ~~or a~~ cent~~er~~ out reach. In ~~the~~ figure 5.9, the Kalman filter for the proprioceptive system predicts the propagation of the system ~~observed through~~ the proprioceptive input. The Kalman filter for the visual predictive system ~~allows us to~~ predict the position and velocity of the cursor visual feedback that represent the hand. Similar to the previous results of the filter, accurate estimates are obtained that lie close~~r~~ to the ground truth of the observation. As observed in the experimental behavior, the proprioceptive, visual and ~~the~~

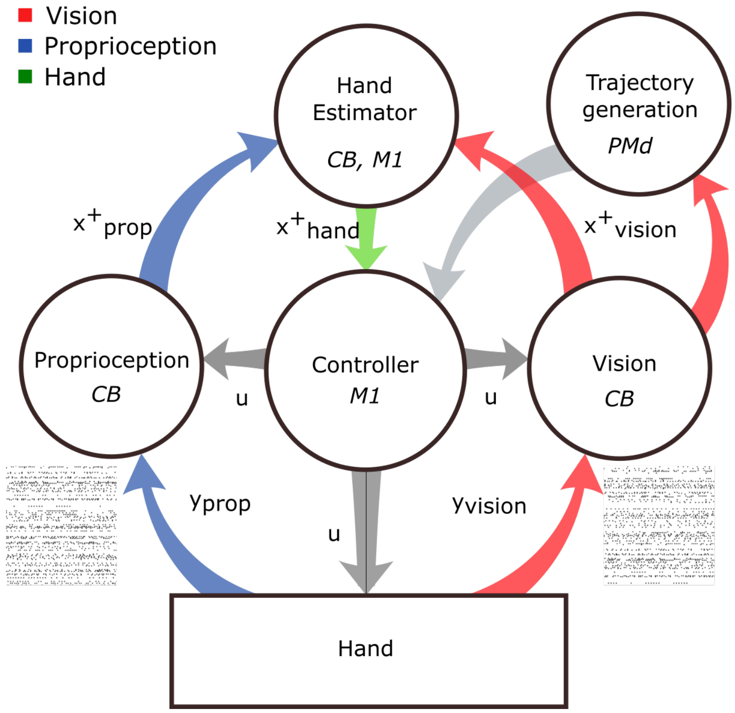

Figure 5.9: **Sensory~~-~~motor control model**: ~~From~~ the two noisy observation of the hand~~,~~ vision and proprioception ~~and~~ the efference copy from the controller two estimates are obtained using the Kalman filters. These are further fused together to know the estimate of the hand position and velocity. A combined movement vector is learn~~t~~ from the hand heading direction and correction realized from the end point error.

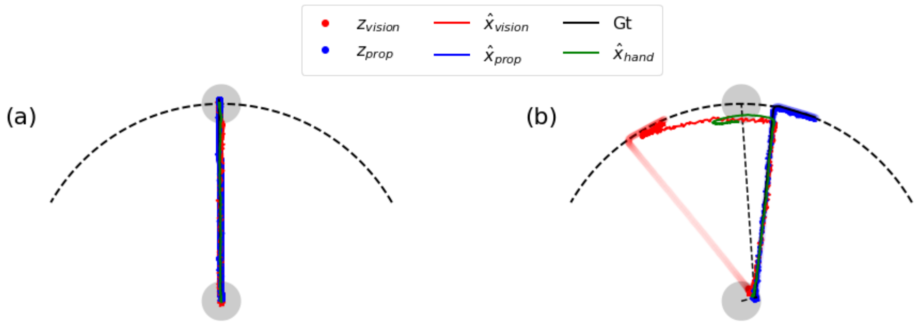

Figure 5.10: **Observations ~~vs E~~stimates**: The visual (red) and proprioceptive (blue) observations and estimate are shown before and after the introduction of visuomotor rotation. (a) Before rotation all the estimates lie on top of each other during the entire trial(b) After rotation, the estimate of ~~the~~ vision splits after the cursor feedback. The hand estimate (green) lies in between the visual and proprioceptive estimates.

resulting combined hand estimate of the model lie~~s~~ on top of each other during the baseline reaches, when there is no rotation involved. The full feedback of the cursor is available, and the estimation relies both on the prediction and the correction from the noisy state observation throughout the entire trial. During the exposure stage of the experiment, where ~~the~~ perturbations are involved, the cursor feedback is made invisible and is provided only at the end of the reach. When the observaations are not available, the error populations are inhibited so that the measurement errors are not added to the system and does not drive any of the learning mechanisms.

~~In the duration of~~ reach where ~~the~~ visual feedback is unavailable, the estimation system completely relies on the prediction until the cursor is observed. Meanwhile the proprioceptive estimate uses ~~boat~~ prediction and the sensory measurements. ~~Indie figure panel B,~~ we can observe that the visual estimate (in red) follows the initial observation of the cursors movement ~~sure~~, and the proprioceptive estimate (in blue) follows the actual hands ground truth position. In accordance with the expected behavior, it can also be seen that the perceived hand estimate lies in between the visual estimate and the proprioceptive estimate. This corresponds to the subject's perception of the hand location, somewhere in between the visual estimate and the proprioceptive estimate, depending upon the ~~relative beliefs~~ of the individual modalities. Once the cursor feedback is available, it can also be seen that the visual estimate is corrected for the new misaligned cursor observation. This in~~-~~turn changes the perceived hand's position and velocity estimate. Hence, within a single trial it is evident ~~daddy~~ biological prediction-perception system works in harmony in generating the expected behavior as observed in experimental results.

Across trials

~~The~~ figure ~~fig~~ shows the trial-by-trial progression of the directional error made by the model. The dotted black line shows the rotational transformation introduced in the experiment. The first 16 trials are the baseline trials, where the visuomotor rotation is absent and the cursor feedback is available throughout the trail. During these trials, ~~bear~~ all the ~~perceptions~~ are in congruence with each other~~,~~ directional error is ~~almost~~ close to 0. From the figure, ~~God the~~ movement trajectories are fairly straight from the origin to the target. This describes a~~n~~ typical reach made ~~by default~~ with ~~complete~~ visual feedback.

Once the rotation is introduced ~~in the~~ in the initial exposure stage, we see a marked difference in the directional error in the experimental data. This behavior is also reproduced by the model, during the rotation block of trials, where the initial directional error is high and the cursor hits the target by an offset. When the cursor feedback is available and the visual estimate is corrected for, we can see highly curved trajectories similar to the experimental observation. This misalignment in the cursor drive the controller to learn a new movement vector as a function of the hand estimate.

With multiple trials of rotational exposure, the model compensates for directional error by changing the trajectory and learning the additional movement vector as a function of the hand location over the course of the trials. The model exemplifies a similar rate of adaptation for the reaches ~~similar to~~ the data. Furthermore, the model's output closely aligns with the actual trajectories from the primate experiments as well. At the end of initial exposure stage, after 80 trials, the directional error is fully compensated, and the hand~~ trajectory~~'s straight path is restored. This behavior of the model is in agreement with the ~~adapted~~ trajectories from the data.

~~a~~fter the exposure stage, a washout stage is introduced. During this stage the rotational elevation is removed and the ~~Michelle~~ feedback is now once again made to align with the proprioceptive measurements. Despite the measurements replicating the original congruence of the feedback in the baseline stage, we observe~~d~~ directional errors in the opposite direction to the experience perturbation. This behavior of the model shows that the sensory motor system has adapted to this transformation and the controller has learn to move the arm to manage the previously seen perturbations. ~~t~~he trend of the directional error across time is also very similar to the experimental data.~~ The removal of the visual motor rotation triggers Disneyland rotation reducing the errors through time.~~ By the end of the washout segment the model has completely learned the rotation and the trajectory is very similar to the ones that are observed during the baseline trials.

This initial implementation has now laid foundation~~s~~ for ~~the~~ biologically plau~~s~~sible ~~framework~~ for the ~~prediciton and~~ perception and ~~the~~ adaptation systems. This model has satisfied many of the ~~expected~~ requirement that we laid out in the beginning for the ~~perception~~ system. Each of the prediction, perception and ~~the~~ adaptive mechanisms are realized in spiking neur~~on~~ network and ~~E~~ach system~~s~~ ~~are~~ modular in structure and also has anatomical significance,

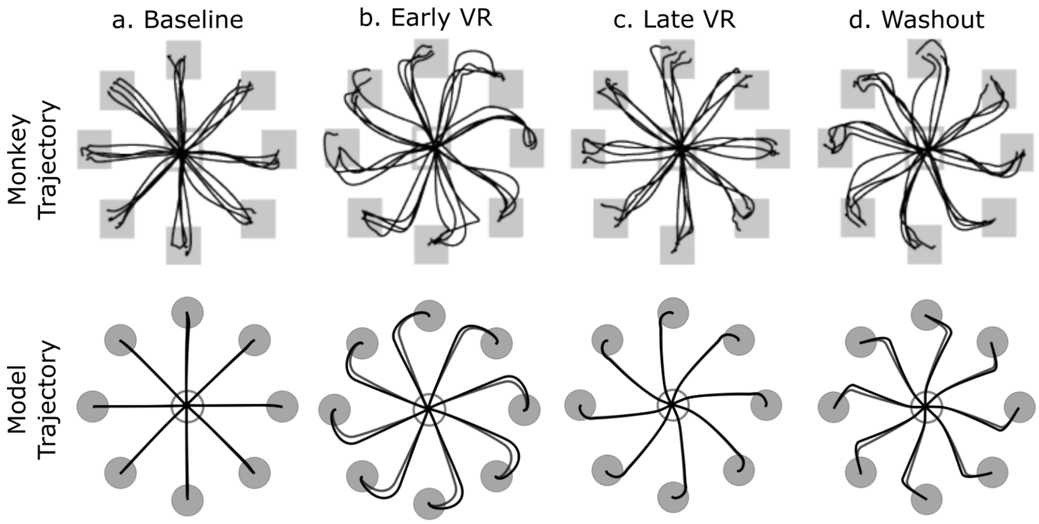

Figure 5.11: **Trajectory comparison** Experiment (top) and model (bottom) during visuomotor rotation: (a) the initial baseline trials without any rotation (b) early adaptation with onset of rotation at 30^o CCW direction (c) later stage of adaptation to rotation (d) washout trials when the rotation is returned to 0^o

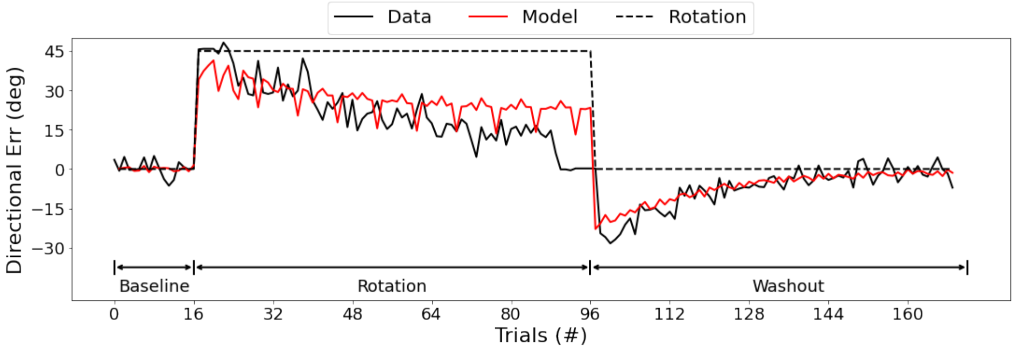

Figure 5.12: **Behavior comparison**: Time course of directional error at threshold of the model's reaches (blue) compared against experimental data (red) (reproduced from the graphs from [Mazzoni and Krakauer, 2006]. The black line shows the angle of rotation of the cursor with respect to the hand location.

which helps us understand the contribution and ~~the working~~ of ~~this~~ structure ~~giving a better explanation for the working of the structure.~~ Furthermore, by reproducing the behavior, the model is also capable of adapting to this complex sensory misalignment and ~~was~~ able to compensate for it in a similar fashion observed in the experimental data both ~~in~~ primates and human ~~trials.~~ This connection of mechanisms, biological plausibility and anatomical significance, collectively establishes ~~a successful~~ modelling of a biological sensory motor control system.

5.5 Limitation of linear system

The ~~given system~~ ~~was~~ able to ~~fit the adaptation~~ by learning to aim for a different target as the cursor appeared at a different location than predicted by the visual estimation system. Although the model ~~was~~ able to replicate the correction for ~~the~~ directional error over the trials, there are a ~~few~~ improvements that can be made to more realistically model the sensory motor prediction system.

1. **Correcting for the visual change**: As soon as the rotational bias is introduced with repeated trials of incorrect visual feedback, the estimation system should predict ~~for~~ the change in dynamics. There is evidence for the cerebellum updating the internal model with experience of errors in predicting the consequences of the command [Imamizu and Kawato, 2009] [Wolpert et al., 1998b]. ~~On the other hand,~~ the visual prediction system here does not evolve over time. The system dynamics ~~is~~ baked into the synapses to help it predict for the given unit mass system. ~~W~~hile the brain is known for adapting to various dynamic systems.

2. **Proper prediction of hand estimate**: By trusting the incoming measurements more than the internal prediction we were able to get away with better visual estimates. But once the visual prediction evolves ~~with the change in~~ observations, ideally we ~~can~~ rely on the prediction more which ~~is~~ not only allows ~~room~~ for biological ~~realism~~ but helps us in aligning the visual estimate ~~follow~~ the actual cursor even in its absence but also have the hand estimate fall between the visual estimate and the proprioceptive estimate in accordance with the actual behavior.

3. **Accommodate for nonlinearity**: On top of continuously evolving to model the plant dynamics, the motor control system handles nonlinearities. The dynamics of ~~an~~ arm, any additional tool, other sensor feedback distortions often ~~function non-linearly~~. As a result, our prediction systems should also handle nonlinear dynamics and not merely work with linear approximat~~es~~ to the plant dynamics.

4. **Work with multiple modalities**: The visual and the proprioceptive stream of information are sensing different quantities, ~~V~~ision gives ~~the~~ information ~~of~~ a virtual ~~”~~end-effector” to control in the task space while the proprioception provides angular position and angular velocities in the

joint angle space. The ideal hand states are then inferred by c~~c~~ombining these two estimates. Hence, for proprioception we need to switch from a ~~common~~ Cartesian space ~~approximation~~ to a joint angle representation and infer the hand location by learning a Jacobian with the help of the measurements. In fact this forms the basis of another kinematic adaptation that is indeed unavoidable for this behavioral task

5. **Further ~~A~~natomical parallels**: ~~a~~lthough the above model had some anatomical significance improvements can be made to detail involvement of ~~some~~ structures. These contributions are missed out in the initial assumptions made for this first iteration. By reinstating those complexities, more cortical structures can be replicated, adding to the ~~holistic~~ reproduction of the anatomical structure in the overall sensory~~-~~motor control system.

To incorporate the above improvements in the following chapter we ~~will~~ discuss a nonlinear adaptive filter and its performance from ~~a~~ perspective of a sensory-motor control system.

Bibliography

Akaike, H. (1969). Fitting autoregressive models for prediction. *Annals of the institute of Statistical Mathematics*, 21(1):243–247.

Alexander, R. M. (1997). A minimum energy cost hypothesis for human arm trajectories. *Biological cybernetics*, 76(2):97–105.

Bedford, F. L. (1993). Perceptual and cognitive spatial learning. *Journal of experimental psychology: Human perception and performance*, 19(3):517.

Bekolay, T., Kolbeck, C., and Eliasmith, C. (2013). Simultaneous unsupervised and supervised learning of cognitive functions in biologically plausible spiking neural networks. In *Proceedings of the annual meeting of the cognitive science society*, volume 35.

Berniker, M. and Kording, K. (2008). Estimating the sources of motor errors for adaptation and generalization. *Nature neuroscience*, 11(12):1454–1461.

Berniker, M. and Penny, S. (2019). A normative approach to neuromotor control. *Biological cybernetics*, 113:83–92.

Bhattacharyya, S. P., Datta, A., and Keel, L. H. (2018). *Linear control theory: structure, robustness, and optimization.* CRC press.

Bliss, T. V. and Collingridge, G. L. (1993). A synaptic model of memory: long-term potentiation in the hippocampus. *Nature*, 361(6407):31–39.

Budišić, M., Mohr, R., and Mezić, I. (2012). Applied koopmanism. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 22(4):047510.

Churchland, M. M., Afshar, A., and Shenoy, K. V. (2006). A central source of movement variability. *Neuron*, 52(6):1085–1096.

Colby, C., Goldberg, M., et al. (1992). The updating of the representation of visual space in parietal cortex by intended eye movements. *Science*, 255(5040):90–92.

Conditt, M. A., Gandolfo, F., and Mussa-Ivaldi, F. A. (1997). The motor system does not learn the dynamics of the arm by rote memorization of past experience. *Journal of Neurophysiology*, 78(1):554–560.

Cueva, C. J. and Wei, X.-X. (2018). Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. *arXiv preprint arXiv:1803.07770*.

Danziger, Z. and Mussa-Ivaldi, F. A. (2012). The influence of visual motion on motor learning. *Journal of Neuroscience*, 32(29):9859–9869.

Denève, S., Alemi, A., and Bourdoukan, R. (2017). The brain as an efficient and robust adaptive learner. *Neuron*, 94(5):969–977.

DeWolf, T., Stewart, T. C., Slotine, J.-J., and Eliasmith, C. (2016). A spiking neural model of adaptive arm control. *Proceedings of the Royal Society B: Biological Sciences*, 283(1843):20162134.

Eliasmith, C. and Anderson, C. H. (2003). *Neural engineering: Computation, representation, and dynamics in neurobiological systems*. MIT press.

Fishbach, A., Roy, S. A., Bastianen, C., Miller, L. E., and Houk, J. C. (2007). Deciding when and how to correct a movement: discrete submovements as a decision making process. *Experimental brain research*, 177:45–63.

Flash, T. and Hogan, N. (1985). The coordination of arm movements: an experimentally confirmed mathematical model. *Journal of neuroscience*, 5(7):1688–1703.

Flash, T. and Sejnowski, T. J. (2001). Computational approaches to motor control. *Current opinion in neurobiology*, 11(6):655–662.

Gandolfo, F., Li, C.-S., Benda, B., Schioppa, C. P., and Bizzi, E. (2000). Cortical correlates of learning in monkeys adapting to a new dynamical environment. *Proceedings of the National Academy of Sciences*, 97(5):2259–2263.

Georgopoulos, A. P., Lurito, J. T., Petrides, M., Schwartz, A. B., and Massey, J. T. (1989). Mental rotation of the neuronal population vector. *Science*, 243(4888):234–236.

Goodbody, S. J. and Wolpert, D. M. (1998). Temporal and amplitude generalization in motor learning. *Journal of Neurophysiology*, 79(4):1825–1838.

Haarmeier, T., Thier, P., Repnow, M., and Petersen, D. (1997). False perception of motion in a patient who cannot compensate for eye movements. *Nature*, 389(6653):849–852.

Haith, A., Jackson, C., Miall, C., and Vijayakumar, S. (2008). Interactions between sensory and motor components of adaptation predicted by a bayesian model. *Workshop on Advances in Computational Motor Control (ACMC 2008)*.

Heuer, H. and Hegele, M. (2008). Adaptation to a nonlinear visuomotor amplitude transformation with continuous and terminal visual feedback. *Journal of Motor Behavior*, 40(5):368–379.

Huang, V. S. and Shadmehr, R. (2007). Evolution of motor memory during the seconds after observation of motor error. *Journal of neurophysiology*, 97(6):3976–3985.

Imamizu, H. and Kawato, M. (2009). Brain mechanisms for predictive control by switching internal models: implications for higher-order cognitive functions. *Psychological Research PRPF*, 73:527–544.

Inoue, M. and Kitazawa, S. (2018). Motor error in parietal area 5 and target error in area 7 drive distinctive adaptation in reaching. *Current Biology*, 28(14):2250–2262.

Ivry, R. B. and Keele, S. W. (1989). Timing functions of the cerebellum. *Journal of cognitive neuroscience*, 1(2):136–152.

Izawa, J. and Shadmehr, R. (2008). On-line processing of uncertain information in visuomotor control. *Journal of Neuroscience*, 28(44):11360–11368.

Johnson, K. O. (2001). The roles and functions of cutaneous mechanoreceptors. *Current opinion in neurobiology*, 11(4):455–461.

Kaiser, E., Kutz, J. N., and Brunton, S. L. (2018). Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proceedings of the Royal Society A*, 474(2219):20180335.

Kang, T., He, J., and Tillery, S. I. H. (2005). Determining natural arm configuration along a reaching trajectory. *Experimental brain research*, 167:352–361.

Kirk, D. E. (2004). *Optimal control theory: an introduction*. Courier Corporation.

Koch, C. and Segev, I. (1998). *Methods in neuronal modeling: from ions to networks*. MIT press.

Körding, K. P. and Wolpert, D. M. (2004). Bayesian integration in sensorimotor learning. *Nature*, 427(6971):244–247.

Krakauer, J. W. (2009). Motor learning and consolidation: the case of visuomotor rotation. In *Progress in motor control*, pages 405–421. Springer.

Krakauer, J. W., Pine, Z. M., Ghilardi, M.-F., and Ghez, C. (2000). Learning of visuomotor transformations for vectorial planning of reaching trajectories. *Journal of neuroscience*, 20(23):8916–8924.

Lackner, J. R. and Dizio, P. (1994). Rapid adaptation to coriolis force perturbations of arm trajectory. *Journal of neurophysiology*, 72(1):299–313.

Lee, C.-H. L., Liu, A., and Chen, W.-S. (2006). Pattern discovery of fuzzy time series for financial prediction. *IEEE Transactions on Knowledge and data Engineering*, 18(5):613–625.

Li, C.-S. R., Padoa-Schioppa, C., and Bizzi, E. (2001). Neuronal correlates of motor performance and motor learning in the primary motor cortex of monkeys adapting to an external force field. *Neuron*, 30(2):593–607.

Liepelt, R., Cramon, D., and Brass, M. (2008). What is matched in direct matching? intention attribution modulates motor priming. *Journal of Experimental Psychology: human perception and performance*, 34(3):578.

Lim, B. and Zohren, S. (2021). Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209.

Lu, D. and Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. *International journal of Remote sensing*, 28(5):823–870.

Lu, L., Meng, X., Mao, Z., and Karniadakis, G. E. (2021). Deepxde: A deep learning library for solving differential equations. *SIAM review*, 63(1):208–228.

Lundquist, C., Sjanic, Z., and Gustafsson, F. (2015). *Statistical Sensor Fusion: Exercises*. Studentlitteratur AB.

MacNeil, J. B., Kearney, R., and Hunter, I. (1992). Identification of time-varying biological systems from ensemble data (joint dynamics application). *IEEE Transactions on Biomedical Engineering*, 39(12):1213–1225.

Mante, V., Sussillo, D., Shenoy, K. V., and Newsome, W. T. (2013). Context-dependent computation by recurrent dynamics in prefrontal cortex. *nature*, 503(7474):78–84.

Manto, M., Bower, J. M., Conforto, A. B., Delgado-García, J. M., Da Guarda, S. N. F., Gerwig, M., Habas, C., Hagura, N., Ivry, R. B., Mariën, P., et al. (2012). Consensus paper: roles of the cerebellum in motor control—the diversity of ideas on cerebellar involvement in movement. *The Cerebellum*, 11:457–487.

Masetty, B., Mirsky, R., Deshpande, A., Mauk, M., and Stone, P. (1989). Is the cerebellum a model-based reinforcement learning agent?

Mazzoni, P. and Krakauer, J. W. (2006). An implicit plan overrides an explicit strategy during visuomotor adaptation. *Journal of neuroscience*, 26(14):3642–3645.

Miall, R., Weir, D. J., Wolpert, D. M., and Stein, J. (1993). Is the cerebellum a smith predictor? *Journal of motor behavior*, 25(3):203–216.

Miall, R. C., Christensen, L. O., and Owen Cain, J. S. (2007). Disruption of state estimation in the human lateral cerebellum. *PLoS biology*, 5(11).

Morgan, R. (2015). Linearization and stability analysis of nonlinear problems. *Rose-Hulman Undergraduate Mathematics Journal*, 16(2):5.

Ohyama, T., Nores, W. L., Murphy, M., and Mauk, M. D. (2003). What the cerebellum computes. *Trends in neurosciences*, 26(4):222–227.

Olberg, R. M., Worthington, A. H., Fox, J. L., Bessette, C., and Loosemore, M. P. (2005). Prey size selection and distance estimation in foraging adult dragonflies. *Journal of comparative physiology A*, 191:791–797.

Padé, H. (1892). Sur la représentation approchée d'une fonction par des fractions rationnelles. In *Annales scientifiques de l'Ecole normale supérieure*, volume 9, pages 3–93.

Prilutsky, B. I. and Zatsiorsky, V. M. (2002). Optimization-based models of muscle coordination. *Exercise and sport sciences reviews*, 30(1):32.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707.

Reuschel, J., Drewing, K., Henriques, D. Y., Rösler, F., and Fiehler, K. (2010). Optimal integration of visual and proprioceptive movement information for the perception of trajectory geometry. *Experimental brain research*, 201(4):853–862.

RobertC, M., Nicoll, and A, R. (1999). Long-term potentiation–a decade of progress? *Science*, 285(5435):1870–1874.

Rushworth, M., Nixon, P., and Passingham, R. (1997). Parietal cortex and movement i. movement selection and reaching. *Experimental brain research*, 117(2):292–310.

Sadtler, P. T., Quick, K. M., Golub, M. D., Chase, S. M., Ryu, S. I., Tyler-Kabara, E. C., Yu, B. M., and Batista, A. P. (2014). Neural constraints on learning. *Nature*, 512(7515):423–426.

Saunders, J. A. and Knill, D. C. (2003). Humans use continuous visual feedback from the hand to control fast reaching movements. *Experimental brain research*, 152(3):341–352.

Scellier, B. and Bengio, Y. (2017). Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Frontiers in computational neuroscience*, 11:24.

Schaeffer, R., Khona, M., and Fiete, I. (2022). No free lunch from deep learning in neuroscience: A case study through models of the entorhinal-hippocampal circuit. *bioRxiv*, pages 2022–08.

Schmid, P. J. (2010). Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28.

Scott, S. H. (2004). Optimal feedback control and the neural basis of volitional motor control. *Nature Reviews Neuroscience*, 5(7):532–545.

Sejnowski, T. J., Churchland, P. S., and Movshon, J. A. (2014). Putting big data to good use in neuroscience. *Nature neuroscience*, 17(11):1440–1441.

Shadmehr, R. and Mussa-Ivaldi, F. A. (1994). Adaptive representation of dynamics during learning of a motor task. *Journal of neuroscience*, 14(5):3208–3224.

Slotine, J.-J. E. and Li, W. (1987). On the adaptive control of robot manipulators. *The international journal of robotics research*, 6(3):49–59.

Stengel, R. F. (1994). *Optimal control and estimation.* Courier Corporation.

Taylor, J. A. and Ivry, R. B. (2011). Flexible cognitive strategies during motor learning. *PLoS computational biology*, 7(3):e1001096.

Telgen, S., Parvin, D., and Diedrichsen, J. (2014). Mirror reversal and visual rotation are learned and consolidated via separate mechanisms: recalibrating or learning de novo? *Journal of Neuroscience*, 34(41):13768–13779.

Thrun, S. (2002). Probabilistic robotics. *Communications of the ACM*, 45(3):52–57.

Todorov, E. and Jordan, M. I. (2002). Optimal feedback control as a theory of motor coordination. *Nature neuroscience*, 5(11):1226–1235.

Uno, Y., Kawato, M., and Suzuki, R. (1989). Formation and control of optimal trajectory in human multijoint arm movement. *Biological cybernetics*, 61(2):89–101.

Vaidyanathan, N., Penny, S., and Berniker, M. (2020). Planned straight or biased to be so? the influence of visual feedback on reaching movements. *Journal of Motor Behavior*, 52(2):236–248.

van Beers, R. J., Wolpert, D. M., and Haggard, P. (2002). When feeling is more important than seeing in sensorimotor adaptation. *Current biology*, 12(10):834–837.

Vaziri, S., Diedrichsen, J., and Shadmehr, R. (2006). Why does the brain predict sensory consequences of oculomotor commands? optimal integration of the predicted and the actual sensory feedback. *Journal of Neuroscience*, 26(16):4188–4197.

Victor, J. D. and Purpura, K. P. (1996). Nature and precision of temporal coding in visual cortex: a metric-space analysis. *Journal of neurophysiology*, 76(2):1310–1326.

Von Helmholtz, H. (1867). *Handbuch der physiologischen Optik*, volume 9. Voss.

Wang, J. and Sainburg, R. L. (2004). Interlimb transfer of novel inertial dynamics is asymmetrical. *Journal of Neurophysiology*, 92(1):349–360.

Warrant, E. (2004). Vision in the dimmest habitats on earth. *Journal of Comparative Physiology A*, 190:765–789.

Wehmeyer, C. and Noé, F. (2018). Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics. *The Journal of chemical physics*, 148(24):241703.

Williams, M. O., Kevrekidis, I. G., and Rowley, C. W. (2015). A data–driven approximation of the koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25:1307–1346.

Wise, S., Moody, S., Blomstrom, K., and Mitz, A. (1998). Changes in motor cortical activity during visuomotor adaptation. *Experimental Brain Research*, 121(3):285–299.

Wolpert, D. M., Ghahramani, Z., and Jordan, M. I. (1995). An internal model for sensorimotor integration. *Science*, 269(5232):1880–1882.

Wolpert, D. M., Goodbody, S. J., and Husain, M. (1998a). Maintaining internal representations: the role of the human superior parietal lobe. *Nature neuroscience*, 1(6):529–533.

Wolpert, D. M., Miall, R. C., and Kawato, M. (1998b). Internal models in the cerebellum. *Trends in cognitive sciences*, 2(9):338–347.