

Optimal Robot Arm Control Using The Minimum Variance Model

Gavin Simmons and Yiannis Demiris

Department of Electrical and Electronic Engineering
Imperial College London
Exhibition Road, London, SW7 2BT
E-mail: {gavin.simmons, y.demiris}@imperial.ac.uk
Web page: <http://www.iis.ee.ic.ac.uk/~y.demiris>

Abstract

Models of human movement from computational neuroscience provide a starting point for building a system that can produce flexible, adaptive movement on a robot. There have been many computational models of human upper limb movement put forward, each attempting to explain one or more of the stereotypical features that characterise such movements.

While these models successfully capture some of the features of human movement, they often lack a compelling biological basis for the criteria they choose to optimise. One that does provide such a basis is the minimum variance model (and its extension Task Optimisation in the Presence of Signal-dependent noise (TOPS)). Here, the variance of the hand position at the end of a movement is minimised, given that the control signals on the arm's actuators are subject to random noise with zero mean and variance proportional to the amplitude of the signal. Since large control signals, required to move fast, would have higher amplitude noise, the speed-accuracy trade-off emerges as a direct result of the optimisation process.

We chose to implement a version of this model that would be suitable for the control of a robot arm, using an optimal control scheme based on the discrete-time linear quadratic regulator. This implementation allowed us to examine the applicability of the minimum variance model to producing human-like movement.

In this paper we describe our implementation of the minimum variance model, both for point-to-point reaching movements and for more complex trajectories involving via-points. We also evaluate its performance in producing human-like movement and show its advantages over other optimisation based models (the well-known minimum jerk and minimum torque-change models) for the control of a robot arm.

1 Introduction

One path to producing flexible, adaptive movement on a robot is to study how humans produce such movements. Computational neuroscience provides a range of theories and models that attempt to explain the common features that characterise human arm movements. Among these features are the straight hand paths and bell-shaped velocity profiles found in point-to-point reaching movements, and the speed-accuracy trade-off formalised

by Fitts Law. Many such models generate their trajectories through the optimisation of some aspect of movement, such as hand velocity or joint torque.

In this paper we describe the implementation of one such model, the minimum variance model, for the production of human-like movement on a robot arm. The trajectories predicted by this theory capture the required external features of human movements and it has a strong biological basis. We have adapted an optimal control scheme to implement this model and demonstrate its applicability to the control of a robot arm.¹ Here, we describe a version of that system extended for a more realistic arm model, and to more complex trajectories through the inclusion of via-points.

We start by describing the characteristic features that define "human-like" movement. We then outline relevant computational theories of human movement that attempt to explain these features. We focus on the minimum variance model, comparing it in simulation to two other well-known models: the minimum jerk and minimum torque-change models. We then describe our implementation of these models using an optimal control scheme. Resulting simulation trajectories for all three models are presented, along with trajectories produced by the minimum variance model controlling a two-link planar robot arm.

2 Human Movement

When performing upper limb movements humans show stereotypical patterns, both between individuals and between trials for the same individual. The following section describes the features we aim to capture on a robot arm. We then look at several computational models of human movement in section 2.2, describing which features they capture.

2.1 Relevant Features

Initially we focus on point-to-point reaching, with extensions to more complex trajectories through the inclusion of via-points discussed in section 3.3.

When reaching between two points humans move their arms to make the path of the hand between the two points roughly straight. Slight curvature does occur, depending on the area of the arm's workspace in which the movement occurs.² These straight movements are smooth: the acceleration profile of the movement contains no discontinuities. This results in a characteristic bell-shaped velocity profile for the movement.^{3, 4}

Point-to-point reaching movements also exhibit an inverse relationship between speed and accuracy, known as Fitts' Law⁵ (1) which states that the faster the movement, the less accurately it will reach the target.

$$T = a + b \left(\log_2 \left(\frac{2A}{W} \right) \right) \quad (1)$$

In (1), T is the movement time, A is the amplitude of the movement, W is the target width, and a and b are coefficients of regression. The term $\frac{2A}{W}$ is known as the index of movement difficulty (ID). Often, a fast inaccurate movement will be followed by short corrective movements to bring the hand back to the target.^{6, 7} This trade-off has been extensively studied in human-computer interaction, where various IDs have been proposed and evaluated against human movements.⁸

Another important aspect of human movement is the relationship between velocity and curvature of a movement, often referred to as the Two-thirds Power Law.^{9, 10, 11} This relationship is formalised by equation (2).

$$V = kR^{(1-\beta)} \quad (2)$$

In (2) V is the tangential velocity of the hand, R is the radius of curvature and k is a proportionality constant. The coefficient β has a value around $\frac{2}{3}$, giving the relation its name. Computational models that seek to capture the features of human movement described so far are outlined in section 2.2.

As well as the spatial aspects of a movement many tasks have temporal requirements, including temporal accuracy. In contrast to the speed-accuracy trade-off for spatial goals, variability of timed actions increases almost linearly with the goal movement time.⁴ When the movement distance increases but movement time stays constant, timing error doesn't increase despite the increased movement velocity. We return to this characteristic of movement in section 3.4.

2.2 Computational Models

A majority of computational neuroscience models that seek to explain these characteristics are based on the assumption that they arise from the optimisation of some criteria by the human motor system.^{12, 13, 14} A notable exception is the equilibrium-point hypothesis and its variations.^{15, 16, 17, 18}

Within the class of optimisation models, it is not clear which aspects of movement should be optimised. A range of criteria have been proposed, such as minimum time of movement, minimum energy expenditure, minimum commanded-torque-change^{19, 20} and the well-known minimum jerk²¹ and minimum torque-change models.² Many of these optimisation criteria have been evaluated against actual human arm movements by previous studies.²²

2.2.1 Minimum Jerk Model

Optimisation criteria, or cost functions, for modelling human movements include both kinematic and dynamic solutions. The minimum jerk model is an example of a kinematic solution. These would seem to be good candidates for explaining human movement, as most targets for reaching are specified in external visual coordinates; it follows that movement planning would also take place in extrinsic coordinates.

It has been suggested that smooth, straight hand paths can be explained if smoothness of movement is an explicit goal of the system.²¹ A good measure of smoothness is the jerk of the movement, defined as the derivative of the Cartesian hand acceleration (the third derivative of the hand position). The cost function of the minimum jerk model is:

$$C_{jerk} = \frac{1}{2} \sum_{n=0}^{T-1} ((\ddot{x}_n)^2 + (\ddot{y}_n)^2) \quad (3)$$

In (3), T is the number of time steps in the movement and \ddot{x} and \ddot{y} represent the jerk of the movement along each axis. Results for this model show a good match between the predicted and most actual trajectories. The hand paths were indeed smooth and straight. Not all features of movement were captured by the minimum jerk model however, including the slight asymmetry in the velocity profile and the speed-accuracy trade-off.

2.2.2 Minimum Torque-change Model

In contrast to the purely kinematic minimum jerk model, the minimum torque-change model² uses a dynamic cost function. While the minimum jerk model successfully captures the general behaviour of reaching movements, it is unlikely that movements are determined independently of dynamic quantities of the arm such as length, load, torque or external force. A cost function was therefore defined² that minimised the change in the torque at the joints:

$$C_{torque-change} = \frac{1}{2} \sum_{n=0}^{T-1} ((\dot{\tau}_{1,n})^2 + (\dot{\tau}_{2,n})^2) \quad (4)$$

T is again the number of time steps in (4), while $\dot{\tau}_{1,n}$ and $\dot{\tau}_{2,n}$ are the derivatives of torque at the shoulder and elbow joints respectively, for each time step n . The hand paths predicted by the model agreed with those predicted by the minimum jerk model in areas close to and in front of the body. Significant differences were predicted however in areas further out and to the side of the body, where the dynamics of the arm differ from those in front of the body. This is a feature of human movement which is captured by models where the hand kinematics are not independent of the physical system used to generate them.^{3, 2}

Experimental results were obtained for human arm trajectories of the movements where the two models differed in their predictions. These results showed clearly that the shape of the hand path was dependent on the area of the workspace where the movement was performed, as predicted by the minimum torque-change model.

The models described so far have focused on producing arm movements with straight hand paths and bell-shaped velocity profiles, but have not addressed the issue of the speed-accuracy trade-off. They also lack a convincing explanation as to why their particular characteristic should be optimised by the motor system.

2.2.3 Minimum Variance Model

It was stated in the previous section that the stereotypical features of human movement hold true between individuals and between repeated movements for the same individual. However, all movements are subject to noise which causes deviations from the desired trajectory. The models mentioned previously do not account for these deviations, or assume them to be negligible. By contrast, the minimum variance model specifically tries to explain these movement disturbances.

Starting from the fact that all neuronal signals are subject to signal-dependent noise (noise whose variance is proportional to the signal amplitude), it follows that such noise on the neuronal motor signals sent to muscle units results in deviations from the desired path. Moving rapidly necessarily requires motor signals with large amplitudes and hence high levels of noise, causing greater deviations. Over the course of the movement these deviations accumulate, leading to inaccuracy of the final arm position and possibly causing failure of the movement goal. Moving as fast as possible is therefore sub-optimal from a goal-achievement point of view. Since different tasks require different levels of spatio-temporal precision, an optimal movement would be one that balances the speed-accuracy trade-off to accomplish the task.

It was therefore proposed that the goal of motor planning is to minimise the variance of the arm's position in the presence of signal-dependent neuronal noise.²³ In this model

the movement time is chosen to achieve a given movement accuracy constraint. To represent goal-directed movements the optimisation criteria is defined in terms of reaching a target position and maintaining it for a post-movement period, during which the summed positional variance should be minimised.²³

$$C_{variance} = \sum_{n=T+1}^{T+N-1} \left((\sigma_x^2)^2 + (\sigma_y^2)^2 \right) \quad (5)$$

As in (3) and (4), T represents the number of time steps in the movement. In (5), the additional term N represents the number of time steps in the post-movement period. σ_x^2 and σ_y^2 are the variances of the hand position along the x- and y-axis respectively. From this definition, the minimum variance model predicts the speed-accuracy trade-off of Fitts' Law. But the other features of human movement described in the previous section are also apparent in trajectories predicted by the model. Despite having no direct criteria for straight, smooth movement, the trajectories show both straight hand paths and bell-shaped velocity profiles. This is a logical consequence of the model, as sudden discontinuous movements require large control signals with correspondingly large noise levels, which would be sub-optimal. Trials with a computational model demonstrate that the Two-thirds Power Law also emerges as a result of minimising the variance of the hand position.²³

As well as reliably capturing the important features of human arm movements, the minimum variance model has other advantages. Unlike other models, it offers a principled explanation as to why the motor system should have evolved to produce movement in this way. Also, the variance of the hand position over repeated trials is a readily observable quantity that can be reliably estimated from visual and proprioceptive information, in contrast to more complex derivative terms such as jerk or joint torque.

3 Implementation

3.1 The Robot Arm

The robot used in this implementation was a two link arm with two rotational degrees-of-freedom (DOF) restricted to movement in a plane. Both DOF were powered by DC motors whose angular velocities were controlled by pulse width modulation. Quadrature encoders on each motor allowed the angular velocities to be returned to the system. Many studies have used a manipulandum to restrict a person to move their arm in a plane, providing data for comparison between the movement of the robot and actual human movement.^{3, 21, 2, 24}

3.2 Optimal control algorithm

In our implementation, the discrete-time linear quadratic regulator (DLQR) optimal control algorithm was used to generate trajectories. The robot arm was represented by the state vector, the composition of which was determined by each model (see sections 3.5, 3.6, and 3.7 below).

After specification of the start coordinates, target coordinates and number of time steps in the movement and post-movement periods, the DLQR algorithm constructs a

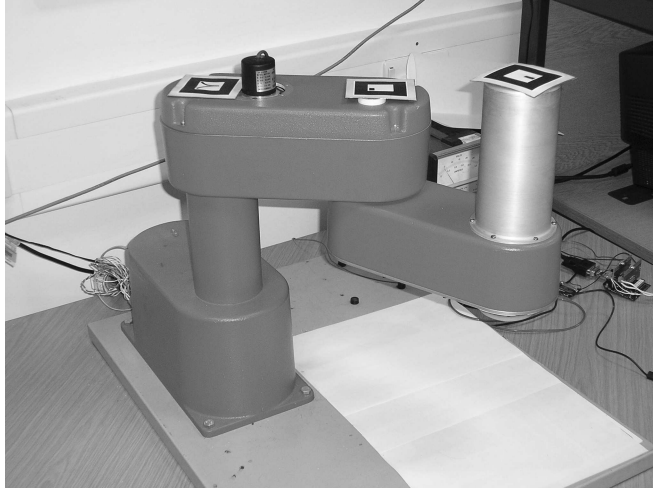


Figure 1: The two link planar robot arm used in these experiments

cost function J for the movement. The matrix form of the cost function, as used for the minimum jerk and minimum torque-change models, is shown below.

$$J = \sum_{n=0}^{T-1} (u'_n R u_n + s'_n Q_n s_n) + s'_T Q_T s_T \quad (6)$$

The minimum variance model requires optimisation of the variance during a post-movement period of length N time steps. The cost function, adapted for optimisation during the post-movement period, therefore takes the form:

$$J = \sum_{n=0}^{T+N-1} (u'_n R u_n) + \sum_{n=0}^{T-1} (s'_n Q_n s_n) + s'_T Q_T s_T + \sum_{n=T+1}^{T+N-1} (s'_n Q_{pm} s_n) \quad (7)$$

In (6) and (7), T is the number of time steps in the movement, N is the number of time steps in the post-movement period, u_n is the control signal at time step n , R is the control signal cost matrix, s_n is the state at time step n , Q_n is the state cost matrix at time step n , Q_T is the target state cost, and Q_{pm} is the state cost during the post-movement period.

The cost-function J is converted into a "cost-to-go" P , which is an estimate of the remaining cost to reach the target at each time step $n = 0, \dots, T, \dots, T + N$. This takes place "off-line" before the movement begins, using Riccati recursion:

$$P_{n-1} = Q_n + (A' P_n A) + \left(A' P_n B (R + B' P_n B)^{-1} B' P_n A \right) \quad (8)$$

The first recursive term is straightforward to find, since the "cost-to-go" for the final step is simply the state cost at that point. From (6), the final state cost for the minimum jerk and minimum torque-change models is Q_T ; therefore the first recursive term for these models is $P_T = Q_T$. For the minimum variance model with its additional post-movement period, the final state cost is Q_{pm} at time step $n = T + N$ (7), making the first recursive term $P_{T+N} = Q_{pm}$.

In (8), A and B are the dynamics matrices used in the state update equation $s_{n+1} = A s_n + B u_n$. The compositions of these matrices are determined by the specific state

representation for each model (see sections 3.5, 3.6 and 3.7). During the movement, the "cost-to-go" P is used to calculate a state feedback gain K (9), which is combined with the current state s_n to produce a motor command u_n :

$$K_n = -(R + B'P_{n+1}B)^{-1} B'P_{n+1}A \quad (9)$$

$$u_n = K_n s_n \quad (10)$$

To represent signal-dependent neuronal noise in the minimum variance model, the control signal is modified by adding a term w_n , randomly drawn from a Gaussian distribution with zero mean and variance proportional to u_n^2 :

$$u_n^{sdn} = u_n + w_n, \quad w_n \sim N(0, k u_n^2) \quad (11)$$

Once calculated, the control signals were used to update the current state and to control the robot arm. Angular velocities were converted into appropriate duty cycles and sent to the arm motors.

3.3 Complex Trajectories and Via-points

While point-to-point reaching movements are the simplest types of movements, humans normally carry out tasks that require more complex trajectories. These complex trajectories are characterised by one or more via-points^{20, 25}.

Each via-point $V^{(i)}$ has two parts: its spatial coordinates and its temporal location $n^{(i)}$ within the course of the movement (12). This is no different from the start or target points, which also have their spatial coordinates and temporal locations ($n = 0$ for the start point and $n = T$ for the target point).

$$V^{(i)} = \{x(n^{(i)}), y(n^{(i)})\}, \quad 0 < n^{(i)} < T \quad (12)$$

However, via-points must be dealt with differently to start and target points, as their temporal position within the course of the movement effects the trajectory as much as their spatial coordinates (Figure 2(a)). Via-points are added to a movement by changing the state cost matrix Q_n (6) at time step $n = n^{(i)}$ to reflect the spatial location of the via-point. The value of $n^{(i)}$ is important, as a via-point that occurs early in a movement will produce a different trajectory from one that occurs at the same spatial coordinates but close to the end of the movement. Despite this, small differences between the temporal locations of via-points (10-100ms) can produce trajectories that are sufficiently similar for most purposes (Figure 2(b)).

Single via-points can be useful for tasks such as obstacle avoidance, while more complex trajectories can be built up using multiple via-points. Via-points can also play a role in combining movements: instead of the target of one learnt movement being the final position, it could be turned into a via-point for a movement that then continues into a second distinct movement.

3.4 Timing Variability

To model timing variability, as described in section 2.1, we add white noise, from a Gaussian distribution with zero mean and variance proportional to T , to the required movement time:

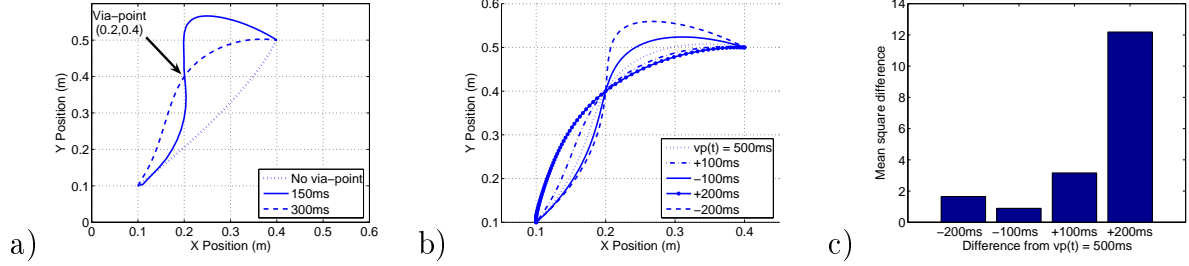


Figure 2: The effect of via-point temporal position: a) Two movements from coordinates (0.1,0.1) to (0.4,0.5) taking 500ms, both of which must pass through the point (0.2,0.4). One must pass through the via-point after 150ms, the other after 300ms: the resulting trajectories are clearly different. b) Average trajectories for 25 movements with differing via-point temporal positions. c) Mean square differences between trajectories with differing via-point temporal positions (see section 4.1).

$$T^{var} = T + w_T, \quad w \sim N(0, kT) \quad (13)$$

This results in movement times that exhibit the same variability as human movement times. If a movement includes via-points, their relative positions within the course of the movement should remain un-changed, in line with the fact that humans can perform learnt motor programs at different speeds while keeping the relative timing of the movement roughly the same.^{4, 26} To take this into account, via-point temporal positions are adjusted by the ratio of the varied movement time T^{var} to the original movement time T :

$$n^{(i),var} = n^{(i)} \cdot \frac{T^{var}}{T} \quad (14)$$

3.5 Minimum Jerk Model

For our implementation of the minimum jerk model, the arm was modelled as a point mass at the position of the hand, attached to a fixed point by two links and constrained to move in a plane. Signal-dependent noise was applied to the control signals used to move the mass and the variance of the hand position during a post-movement period was minimised. Start and target points were given in terms of a two-dimensional Cartesian frame-of-reference whose origin was at the fixed shoulder joint. The state vector s_{jerk} for this model consisted of the hand position (x, y) , and the hand velocities (\dot{x}, \dot{y}) , accelerations (\ddot{x}, \ddot{y}) and jerks (\dddot{x}, \dddot{y}) :

$$s_{jerk} = [x \quad y \quad \dot{x} \quad \dot{y} \quad \ddot{x} \quad \ddot{y} \quad \dddot{x} \quad \dddot{y}]' \quad (15)$$

The cost function followed the form of (3) in section 2.2.1 and (6) in section 3.2, with Q_n penalising the jerk at each time step. Q_T differed only in that it contained information about the target and a term to bring the movement to a halt.

In the standard formulation of the optimal control problem, the jerk would be considered the control input and would therefore not be included in the state vector. However, in the minimum variance model the term to be optimised is not the control input and must therefore be included explicitly in the state vector. To maintain consistency between the models, we have included the optimised term for the minimum jerk model in the state

vector. This makes no difference to the dynamics of the system, which are shown in equation (16) for the x-axis components of the state. The same dynamics are used for the y-axis components. These full system dynamics are put into matrix form suitable for use in the state update equation $s_{n+1} = As_n + Bu_n$ (see section 3.2), giving the dynamics matrices A_{jerk} and B_{jerk} (17). In the following equations Δt is the step time and u_n is the control signal at time step n .

$$\begin{aligned} x_{n+1} &= x_n + \Delta t \cdot \dot{x}_n \\ \dot{x}_{n+1} &= \dot{x}_n + \Delta t \cdot \ddot{x}_n \\ \ddot{x}_{n+1} &= \ddot{x}_n + \Delta t \cdot u_n \\ \ddot{x}_n &= \frac{\ddot{x}_{n+1} - \ddot{x}_n}{\Delta t} = u_n \end{aligned} \quad (16)$$

$$A_{jerk} = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \Delta t & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad B_{jerk} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \Delta t & 0 \\ 0 & \Delta t \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (17)$$

A major consideration for using this model on a robot arm was the fact that Cartesian positions and velocities had to be converted into angles and angular velocities at each time step to control the robot arm. This was not difficult or computationally expensive with a two-link planar arm, but would have significantly slowed the system with a more complicated arm model, as the inverse kinematics function itself would have been correspondingly more complicated. Inverse kinematic functions are also one-to-many functions, meaning that the arm configuration calculated at each time step would have to be checked against the configuration at the previous time step to ensure joint angle continuity.

3.6 Minimum Torque-change Model

Given the drawbacks of the point-mass arm used to implement the minimum jerk model, a more realistic arm was used when implementing the minimum torque change model. Instead of controlling a point-mass at the position of the hand, the joint angles of the arm were controlled directly. The position of the hand was calculated at each time step using the forward kinematics, which are much less computationally expensive than the inverse kinematics. They are also many-to-one functions, so there was no ambiguity about the position of the hand. The start and target coordinates were specified using a Cartesian frame-of-reference as before, but these were converted into starting and target joint angles using a single instance of the inverse kinematics before the "on-line" processing of the movement began.

The state vector $s_{torque-change}$ of this model consisted of the joint angles at the shoulder (θ_1) and elbow (θ_2), their angular velocities and accelerations, the torque at each joint τ_1 and τ_2 , the corresponding torque-change and the Cartesian position of the hand (x, y):

$$s_{torque-change} = [\theta_1 \quad \theta_2 \quad \dot{\theta}_1 \quad \dot{\theta}_2 \quad \ddot{\theta}_1 \quad \ddot{\theta}_2 \quad \tau_1 \quad \tau_2 \quad \dot{\tau}_1 \quad \dot{\tau}_2 \quad x \quad y] \quad (18)$$

With this state vector, the joint angles and angular velocities were directly accessible from the state vector at each time step. These could be converted directly to motor

velocities for a robot arm driven by DC-motors. In a similar way to the minimum jerk implementation, the cost function followed the form of (6) in section 3.2, with terms to penalise the torque-change rather than the jerk ((4) in section 2.2.2).

Also following the minimum jerk implementation, the optimised term was included explicitly in the state vector, maintaining consistency with the implementation of the minimum variance model. Again this did not affect the dynamics of the system, given in equation (19) for both the shoulder and elbow joints, represented by θ . The torques and Cartesian hand positions were calculated from the joint angles using the standard forward dynamics and forward kinematics functions respectively². The dynamics matrices $A_{torque-change}$ and $B_{torque-change}$ are given by equation (20).

$$\begin{aligned}\theta_{n+1} &= \theta_n + \Delta t \cdot \dot{\theta}_n \\ \dot{\theta}_{n+1} &= \dot{\theta}_n + \Delta t \cdot \ddot{\theta}_n \\ \ddot{\theta}_{n+1} &= \ddot{\theta}_n + \Delta t \cdot u_n\end{aligned}\tag{19}$$

$$A_{torque-change} = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \Delta t & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \Delta t & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, B_{torque-change} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \Delta t & 0 \\ 0 & \Delta t \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix}\tag{20}$$

While more appropriate for controlling the robot arm than the minimum jerk model, this model still lacks an accuracy constraint. It has the added disadvantage that the criteria it optimises is difficult to measure and calculate compared to a quantity such as jerk, requiring joint angles, angular velocities and accelerations to generate the torque, which then requires differentiation to find the torque-change.

3.7 Minimum Variance Model

The minimum variance model has been implemented in various forms. The original formulation²³ used a combination of muscle and skeletal models for the arm, combined with optimisation of cubic splines to determine the minimum variance trajectory. Other approaches include the combination of a minimum jerk trajectory generator and a recurrent network to produce the required movements²⁷ and a Kalman filter method with a muscle model, used to find examine feedback control laws.²⁸ While each of these were appropriate for their purposes, they were not directly applicable to the control of a robot arm for producing human-like movement.

The direct joint control used for the minimum torque-change model was also used here. Unlike the other two models, the modified cost function (7) was used in the DLQR and signal-dependent noise was applied to the control signals (11). Following the principle of minimising the variance only during the post-movement period, Q_n was set to zero for the duration of the movement. Q_T contained information about the target, while Q_{pm} was the only term to penalise the variance of the hand position.

The state vector $s_{variance}$ for this model consisted of the two joint angles of the arm θ_1 and θ_2 , their angular velocities, the Cartesian coordinates of the hand (x, y) , and the variances in the hand position for each axis (σ_x^2, σ_y^2) :

$$s_{variance} = [\theta_1 \quad \theta_2 \quad \dot{\theta}_1 \quad \dot{\theta}_2 \quad x \quad y \quad \sigma_x^2 \quad \sigma_y^2]'$$

This state vector brings the implementation of the minimum variance model close to previous formulations^{23, 28}, but still without modelling the unnecessary muscle function. That is, it attempts to minimise an extrinsic value (variance of the hand position) by controlling an intrinsic value (joint angles). The trajectories that come from controlling the joint angles² are coupled with an easily observable kinematic optimisation criterion.

The dynamics of the system were similar to those of the minimum torque-change model, given in equations (19) and (20), with correspondingly fewer terms to match the state vector of the minimum variance model. The Cartesian hand position was calculated from the joint angles using the standard forward kinematics equations. The hand variances were calculated using the common textbook definition, given in equation (21), where n is the current step, x is the x-coordinate of the hand position and \bar{x} is the mean of the x-coordinate of the hand position to time step n .

$$\sigma_{x,n}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 \quad (21)$$

An interesting aspect of the minimum variance model over other optimisation models is that strict boundary conditions, when the errors in position, velocity and acceleration of the start and end points are required to become strictly zero, are no longer necessary. Since the final state will vary from the target state due to the noise, the goal of the algorithm is not to reach the exact target point at the end of the movement, but rather to move the hand to the target with a certain level of task-dependent variance.²⁷ This flexibility means that velocity constraints at the target position can also be relaxed, allowing movements such as catching a ball to be performed in the same way as reaching to a static target.

4 Results

4.1 Movement Analysis

The movements produced by our implementation of the minimum variance model were evaluated in a number of ways. Qualitative analysis of the resulting trajectories ensured that the hand paths were indeed straight and that the velocity profiles had the required shape.

Quantitative analysis was used to show the minimisation of variance and the effects of the speed-accuracy trade-off. Accuracy was specified in terms of the standard deviation of the final hand position. This was recorded over a number of repeated movements between the same points, for different instructed movement times.

For more complex trajectories involving via-points a metric covering the whole of the trajectory was required. A combined segmentation method²⁹ was used with time-scaling to find the mean square difference $d(\alpha, \beta)$ between repeated trajectories to further show the speed-accuracy trade-off.

$$d(\alpha, \beta) = \sum_{n=0}^{\min(T_\alpha, T_\beta)} \sum_{j=1}^J (\alpha_n^{(j)} - \beta_n^{(j)})^2 \quad (22)$$

This metric works in joint space to compensate for differences in arm lengths between individuals. In (22), α and β are the trajectories to be compared, of length T_α and T_β time steps respectively and each consisting of $j = 1, \dots, J$ joint angles. The combined segmentation method is flexible enough to be used with single movements (with or without via-points) and with combinations of movements performed in a sequence.

4.2 Minimum Jerk Model

Figure 3 shows simulation hand paths and velocity profiles for ten movements performed between similar points in one area of the arm's workspace. As can be seen in Figure 3(a), the movements predicted by the minimum jerk model fit the criteria for straight hand trajectories described in section 2.1. The trajectories were smooth with characteristic bell-shaped velocity profiles²¹ (Figure 3(b)). However, they lack the curvature typical of human movements and they do not change according to the area of the workspace where the movement occurs (see Figure 8 in section 4.4 below). As stated previously, the minimum jerk model also lacks an accuracy constraint.

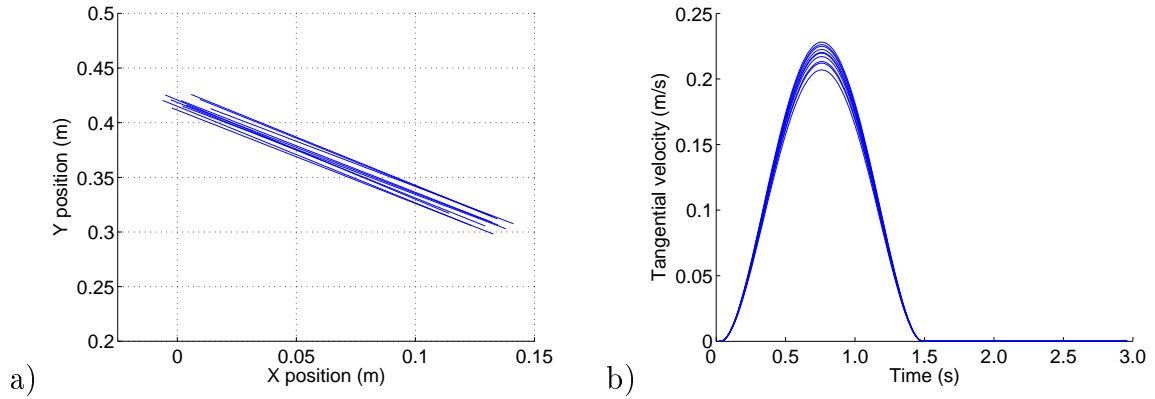


Figure 3: Ten typical trajectories produced by the minimum jerk model in simulation: a) Hand paths; b) Tangential velocity profiles

4.3 Minimum Torque-change Model

Figure 4 shows hand paths and velocity profiles for ten movements carried out with the minimum torque-change model. These movements are between similar points, and occur in the same area of the arm's workspace, as those movements carried out with the minimum jerk model and shown in Figure 3.

The trajectories produced by the dynamic minimum torque-change model are closer to those produced by humans than the trajectories of the purely kinematic minimum jerk model: they are not perfectly straight but are instead slightly curved, clearly shown in Figure 4(a). The hand paths are still smooth however, as demonstrated by the velocity

profiles shown in Figure 4(b) which match closely with those of the minimum jerk model in Figure 3(b).

The minimum torque-change trajectories closely resemble those of the minimum jerk model in areas of the workspace close to and in front of the body (Figures 3(a) and 4(a)). In areas away from the body, where the dynamics are different, the trajectories are more curved, matching results of human arm movements.² This is best seen in Figure 8, where a movement between two points at the extent of the arm’s reach was performed with each of the three models. While the velocity profiles are broadly similar, the hand path for the minimum jerk model clearly differs from that of the other two.

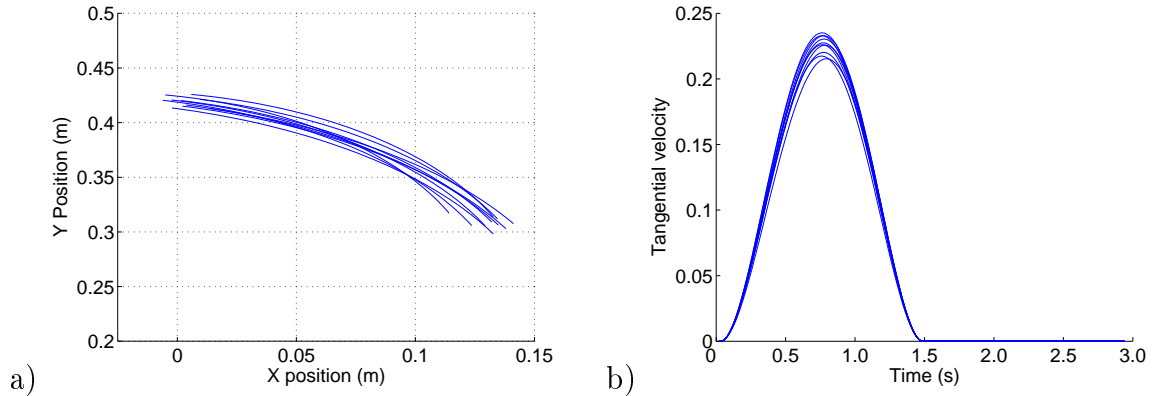


Figure 4: Ten typical trajectories produced by the minimum torque-change model in simulation: a) Hand paths; b) Tangential velocity profiles

4.4 Minimum Variance Model

The trajectories predicted by this version of the model exhibited the required features of human movement. For point-to-point reaching movements without via-points the trajectories were slightly curved in the same way as human arm movements (Figures 5(a) and 6(a)), matching those of the dynamic minimum torque-change model without an explicitly dynamic cost function.

The simulation trajectories of Figure 5 are very similar to those of the minimum torque-change model, although the velocity profiles differ slightly. The same movements carried out by the robot arm differ slightly from the simulation results, but result in similar slightly curved hand paths and the required velocity profiles (Figure 6).

In addition to roughly straight smooth movements, the addition of signal-dependent noise on the control signals results in changing end-point variance for different required movement times. Figure 7 clearly shows the decrease in end-point variance with increasing movement time predicted by the model. This speed-accuracy trade-off matches that of Fitts’ Law in human movement.

As discussed above, Figure 8 shows a movement between two points carried out by all three models. The minimum torque-change model and minimum variance model hand paths match closely. This is a result of the under-lying nature of the minimum variance model: the cost function minimises a kinematic variable (hand positional variance), but the resulting trajectory depends on the arm dynamics because the computational process is dynamic.²⁴

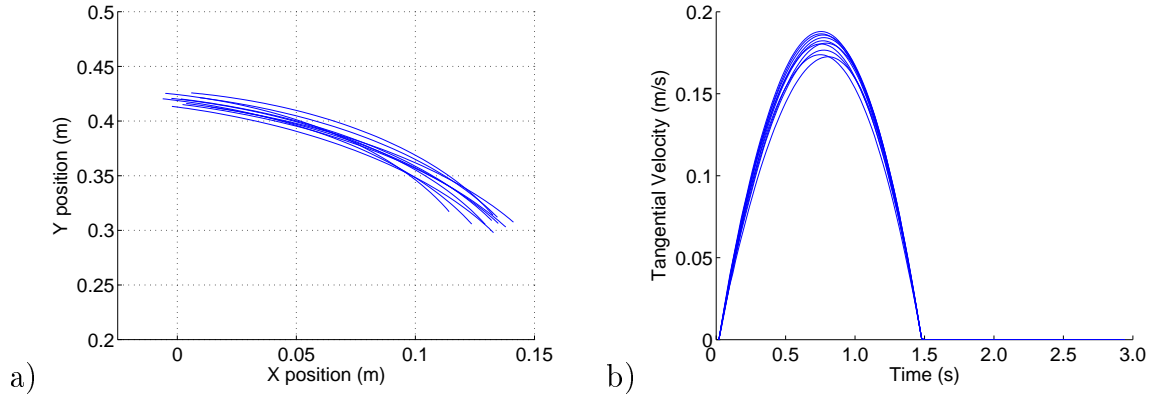


Figure 5: Ten typical trajectories produced by the minimum variance model in simulation: a) Hand paths; b) Tangential velocity profiles.

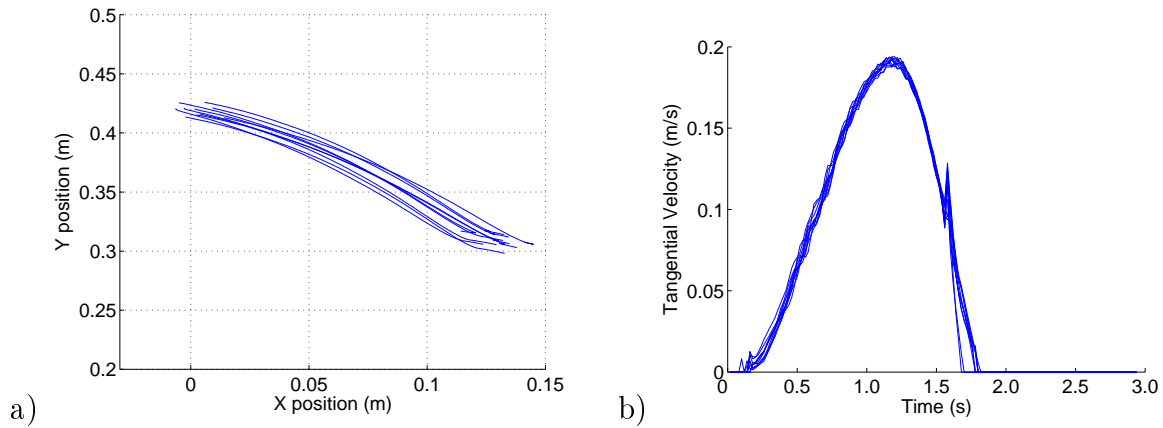


Figure 6: Ten typical trajectories produced by the minimum variance model on the robot arm: a) Hand paths; b) Tangential velocity profiles of the hand.

5 Discussion

Computational models of human movement vary in the observable characteristics of such movement that they capture and in their applicability to controlling a robot arm. In this paper we have focused on the minimum variance model and demonstrated that not only does it capture the common characteristics of human arm movements, but that it is also suitable for implementation on a robotic platform, allowing a robot arm to move in a human-like manner.

We have described our implementation of the minimum variance model and compared the resulting trajectories with those produced by two other well-known models. These models capture some, but not all, of the common features of human movement. Their criteria of jerk and torque-change are also less straightforward to calculate and optimise than the readily observable quantity of hand positional variance.

A useful application of this scheme for producing human-like movement on a robot arm is for programming by demonstration. Coupling of an imitation model that uses a common system for motor production and movement recognition^{30, 31}, and the optimal control scheme described here allows the system to recognise, learn and imitate human

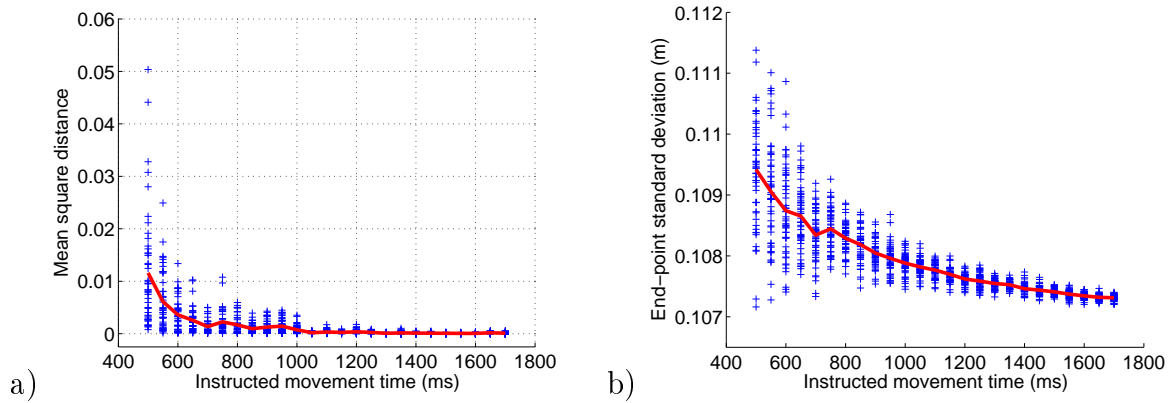


Figure 7: The speed-accuracy trade-off exhibited by the minimum variance model: A number of movements between the same points were made with increasing movement times, and the resulting trajectories compared in different ways. a) The mean square distance between 10 trajectories was calculated (see section 4.1). Each trajectory was compared against each of the others, giving $(10^2 - 10) \div 2 = 45$ comparisons for each instructed movement time; b) The standard deviation of the end position of the hand for 50 repeated movements was calculated. Both measures show an increase in accuracy with increasing movement time.

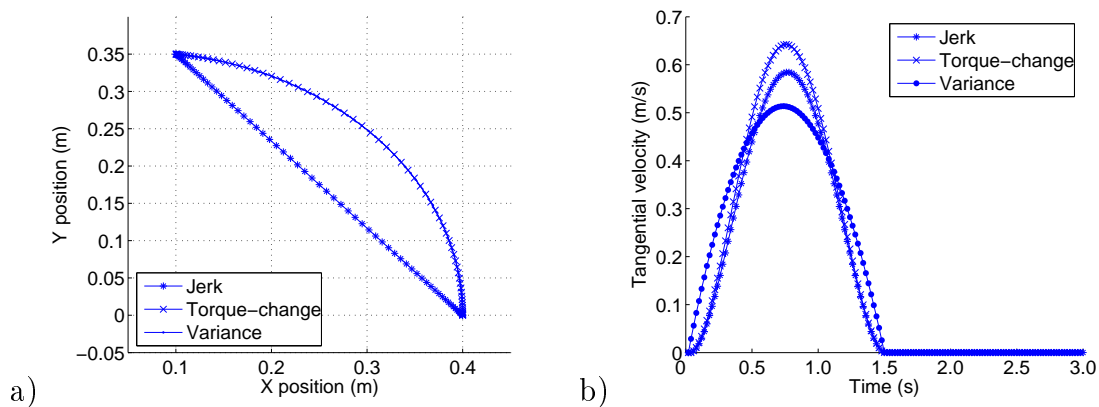


Figure 8: Three movements between the same points, one carried out with the minimum jerk model, one with other with the minimum torque-change model and one with the minimum variance model. The velocity profiles are all very similar (b) but the hand path of the minimum jerk differs considerably from those of the minimum torque-change and minimum variance models (a), reflecting the changes to movement in different areas of the arm's workspace.

movements and tasks³².

Acknowledgements

This work has been supported through a Doctoral Training Award from the UK's Engineering and Physical Sciences Research Council (EPSRC). The authors wish to thank

Anthony Dearden, Bassam Khadhour, Matthew Johnson and Paschalis Veskos of the Biologically inspired Autonomous Robots Team (BioART) at Imperial College.

References

- [1] G. Simmons and Y. Demiris. Biologically inspired optimal robot arm control with signal-dependent noise. *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2004.
- [2] Y. Uno, M. Kawato, and R. Suzuki. Formation and control of optimal trajectory in human multijoint arm movement: Minimum torque-change model. *Biological Cybernetics* 61:89–101, 1989.
- [3] P. Morasso. Spatial control of arm movements. *Experimental Brain Research* 42:223–227, 1981.
- [4] R. A. Schmidt. *Motor Learning and Performance: From Principles to Practice*. 1991.
- [5] P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology* 47(6):381–391, 1954.
- [6] A. H. Fagg, L. Zelevinsky, A. G. Barto, and J. C. Houk. Using crude corrective movements to learn accurate motor programs for reaching. *Extended Abstracts of NIPS '97 Workshop: Can Artificial Cerebellar Models Compete to Control Robots?*, chapter 6, pp. 20–24, 1997.
- [7] A. H. Fagg, A. G. Barto, and J. C. Houk. Learning to reach via corrective movements. *Proceedings of the Tenth Yale Workshop on Adaptive and Learning Systems*, pp. 179–185, 1998.
- [8] J. Accot and S. Zhai. Beyond Fitts' law: Models for trajectory-based HCI tasks. *Proceedings of CHI '97*, pp. 295–302, 1997.
- [9] F. Lacquaniti, C. A. Terzuola, and P. Viviani. The law relating kinematic and figural aspects of drawing movements. *Acta Psychologica* 54:115–130, 1983.
- [10] P. Viviani and R. Schneider. A developmental study of the relationship between geometry and kinematics in drawing movements. *Journal of Experimental Psychology: Human Perception and Performance* 17:198–218, 1991.
- [11] M. J. E. Richardson and T. Flash. Comparing smooth arm movements with the two-thirds power law and the related segmented-control hypothesis. *The Journal of Neuroscience* 22(18):8201–8211, September 2002.
- [12] G. E. Loeb, W. Levine, and J. He. Understanding sensorimotor feedback through optimal control. *Symposium on Quantitative Biology* 55:791–803, 1990.
- [13] T. Flash and T. J. Sejnowski. Computational approaches to motor control. *Current Opinion in Neurobiology* 11:655–662, 2001.

- [14] Z. Luo, M. Svinin, K. Ohta, T. Odashima, and S. Hosoe. On optimality of human arm movements. *Proceedings of the IEEE International Conference on Robotics and Biomimetics 2004*, p. 447, August 2004.
- [15] E. Bizzi, F. A. Mussa-Ivaldi, and S. Giszter. Computations underlying the execution of movement: A biological perspective. *Science* 253:287–291, July 1991.
- [16] H. Gomi and M. Kawato. Equilibrium-point control hypothesis examined by measured arm stiffness during multijoint movement. *Science* 272:117–120, April 1996.
- [17] F. A. Mussa-Ivaldi. Motor primitives, force-fields and the equilibrium point theory. *From Basic Motor Control to Functional Recovery*, pp. 392–398, 1999.
- [18] M. R. Hinder and T. E. Milner. The case for an internal dynamics model versus equilibrium point control in human movement. *Journal of Physiology* 549(3):953–963, 2003.
- [19] E. Nakano, H. Imamizu, R. Osu, Y. Uno, H. Gomi, T. Yoshioka, and M. Kawato. Quantitative examinations of internal representations for arm trajectory planning: Minimum commanded torque change model. *Journal of Neurophysiology* 81(5):2140–2155, 1999.
- [20] Y. Wada and M. Kawato. A via-point time optimization algorithm for complex sequential trajectory formation. *Neural Networks* 17:353–364, 2004.
- [21] T. Flash and N. Hogan. The coordination of arm movements: An experimentally confirmed mathematical model. *Journal of Neuroscience* 5(7):1688–1703, July 1985.
- [22] J. G. Hale and F. E. Pollick. Biomimetic synthesis for the upper limb based on human motor production. *7th International Conference on Simulation of Adaptive Behaviour*, August 2002.
- [23] C. M. Harris and D. M. Wolpert. Signal-dependent noise determines motor planning. *Nature* 394:780–784, August 1998.
- [24] H. Miyamoto, E. Nakano, D. M. Wolpert, and M. Kawato. TOPS (Task Optimization in the Presence of Signal-dependent noise) model. *Systems and Computers in Japan* 35(11):48–58, 2004.
- [25] H. Miyamoto, J. Morimoto, K. Doya, and M. Kawato. Reinforcement learning with via-point representation. *Neural Networks* 17:299–305, 2004.
- [26] S. J. Goodbody and D. M. Wolpert. Temporal and amplitude generalization in motor learning. *Journal of Neurophysiology* 79:1825–1838, 1998.
- [27] H. Miyamoto, D. M. Wolpert, and M. Kawato. Computing the optimal trajectory of arm movement: The TOPS (Task Optimization in the Presence of Signal-dependent noise) model. *Biologically Inspired Robot Behaviour Engineering*, vol. 109, chapter 14, pp. 395–416. Springer-Verlag, Studies In Fuzziness And Soft Computing, 2003.
- [28] E. Todorov and M. I. Jordan. Optimal feedback control as a theory of motor coordination. *Nature Neuroscience* 5(11):1226–1235, November 2002.

- [29] M. Pomplun and M. Mataric. Evaluation metrics and results of human arm movement imitation. *Proceedings of the 1st IEEE-RAS International Conference on Humanoid Robotics*, 2000.
- [30] Y. Demiris and G. Hayes. Imitation as a dual-route process featuring predictive and learning components: A biologically-plausible computational model. *Imitation in Animals and Artifacts*, chapter 13. MIT Press, 2002.
- [31] Y. Demiris and M. Johnson. Distributed, predictive perception of actions: A biologically inspired robotics architecture for imitation and learning. *Connection Science Journal* 15(4):231–243, December 2003.
- [32] G. Simmons and Y. Demiris. Imitation of human demonstration using a biologically inspired modular optimal control scheme. *Proceedings of the 2004 IEEE-RAS/RSJ International Conference on Humanoid Robots*, 2004.