MOTOR LEARNING AND MEMORY (2)

NONLINEAR ADAPTIVE CONTROL

- learning a dynamic model which is valid for arbitrary movements

  $$\boldsymbol{\tau} = \boldsymbol{\Psi}(\mathbf{q_u}, \dot{\mathbf{q}}_\mathbf{u}, \ddot{\mathbf{q}}_\mathbf{u})\,\mathbf{p} + \mathbf{K}\boldsymbol{\varepsilon}\,,\ \boldsymbol{\varepsilon} \equiv \mathbf{e} + \kappa\dot{\mathbf{e}}\,,\ \mathbf{e} \equiv \mathbf{q_u} - \mathbf{q}$$

- adapt the feedforward command $\boldsymbol{\Psi}\mathbf{p}$ to minimise the feedback component $\mathbf{K}\boldsymbol{\varepsilon}$

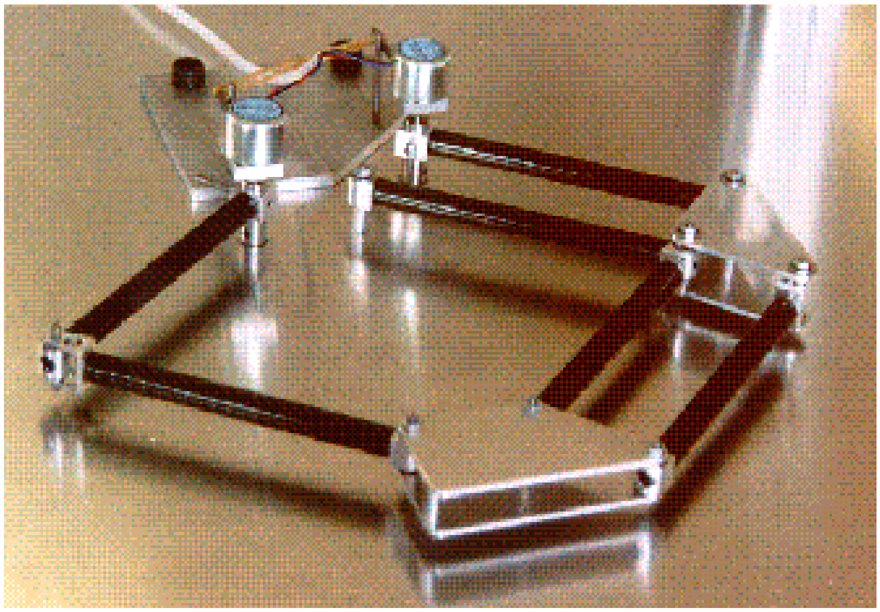
- gradient descent of the cost function

  $$V \equiv \frac{1}{2}\boldsymbol{\tau}_{FB}^T\boldsymbol{\tau}_{FB}\,, \qquad \boldsymbol{\tau}_{FB} \equiv \mathbf{K}\boldsymbol{\varepsilon}$$

- yields the **adaptation law**

  $$\mathbf{p}^{k+1} = \mathbf{p}^k - \alpha\frac{dV}{d\mathbf{p}} = \mathbf{p}^k + \alpha\boldsymbol{\Psi}^T\boldsymbol{\tau}_{FB}\,, \qquad \alpha > 0$$

NONLINEAR ADAPTIVE CONTROL



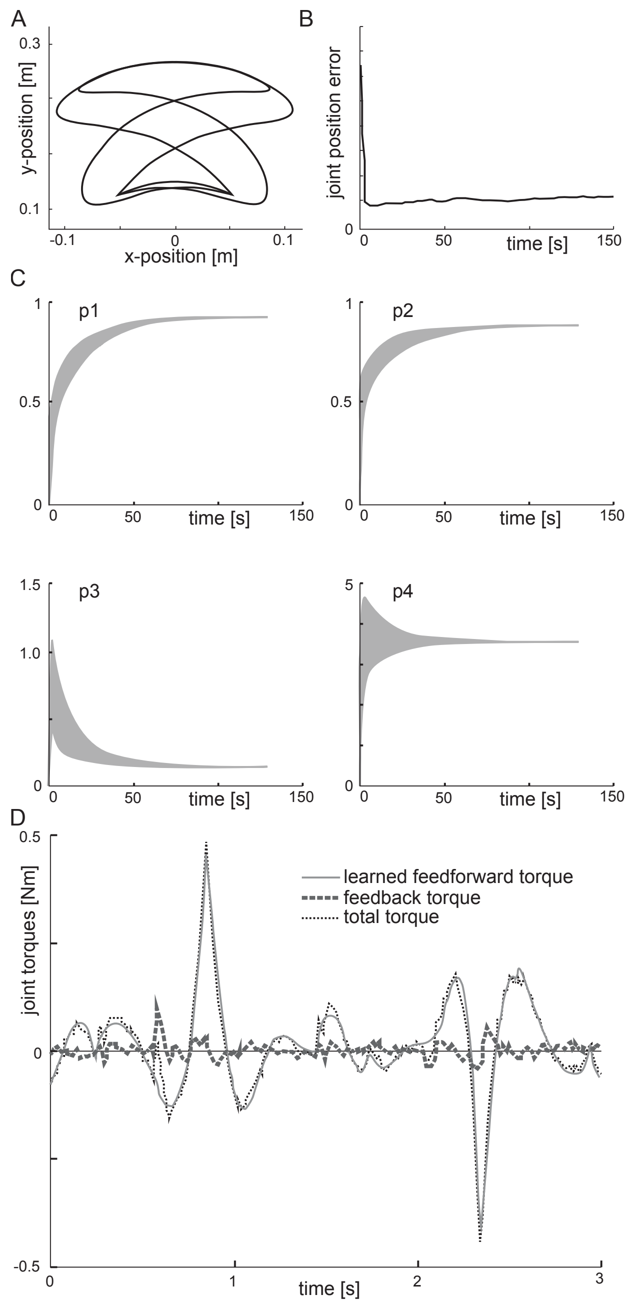
$$\boldsymbol{\tau} = \boldsymbol{\Psi}\,\mathbf{p}\,, \quad \mathbf{p} = (p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8)^T$$

$$\boldsymbol{\Psi} = \begin{bmatrix} s_{12}\dot{q}_2^2 - c_{12}\ddot{q}_2 + \ddot{q}_1 & s_{12}\dot{q}_1^2 - c_{12}\ddot{q}_1 + \ddot{q}_2 \\ \frac{3}{2}\left(s_{12}\dot{q}_2^2 - c_{12}\ddot{q}_2\right) + \frac{11}{4}\ddot{q}_1 & \frac{3}{2}\left(s_{12}\dot{q}_1^2 - c_{12}\ddot{q}_1\right) + \frac{7}{4}\ddot{q}_2 \\ \ddot{q}_1 & \ddot{q}_2 \\ \ddot{q}_1 & 0 \\ \dot{q}_1 & 0 \\ \mathrm{sign}(\dot{q}_1) & 0 \\ 0 & \dot{q}_2 \\ 0 & \mathrm{sign}(\dot{q}_2) \end{bmatrix}^T$$

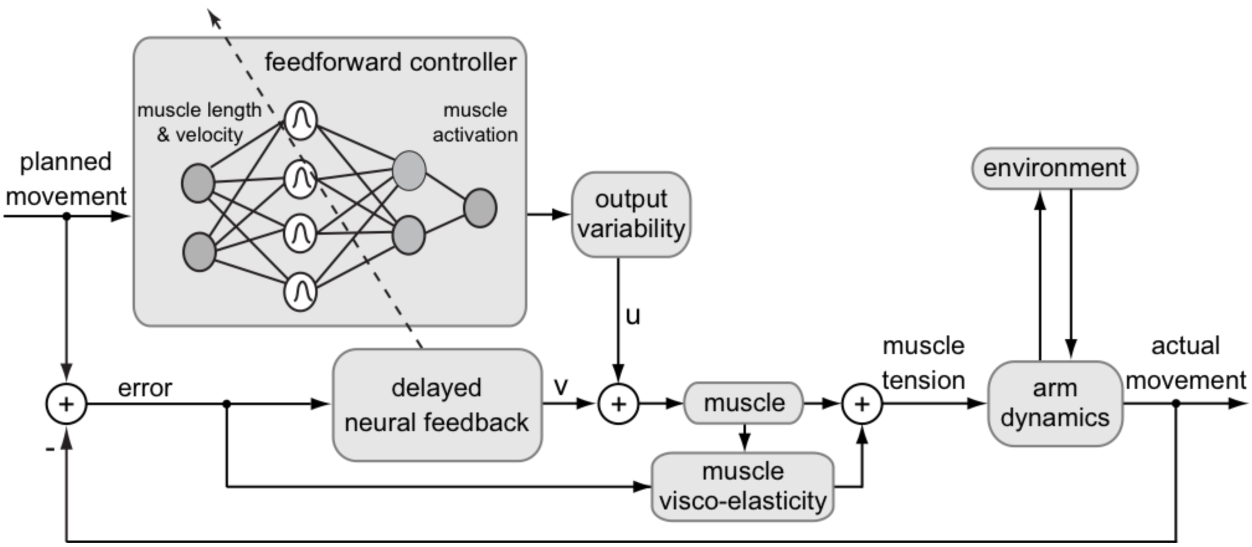
$$s_{12} \equiv \sin(q_1 + q_2) \quad c_{12} \equiv \cos(q_1 + q_2)$$

to identify the parameters $\mathbf{p}$ using nonlinear adaptive control

NONLINEAR ADAPTIVE CONTROL



RBF NEURAL NETWORK
FEEDFORWARD MODEL



$$
\begin{aligned}
\mathbf{u} &= \mathbf{W}\,\mathbf{\Phi}(\mathrm{s})\,, \quad \mathrm{s} \equiv (\mathbf{q}, \dot{\mathbf{q}}) \\
\mathbf{\Phi} &= (\phi_1, \phi_2, \ldots, \phi_N)^T, \quad \phi_j(\mathrm{s}) = \exp\left[\frac{||\mathrm{s} - \mathrm{s}_j||^2}{2\,\sigma_j^2}\right]
\end{aligned}
$$

N **neurones** with **centres** $\mathrm{s}_j \equiv (\mathbf{q}_j, \dot{\mathbf{q}}_j)$ and **activation fields** σ_j

RBF NEURAL NETWORK FEEDFORWARD MODEL

$$\mathbf{\Psi p} \equiv \mathbf{W\Phi}\,, \qquad \mathbf{W} = (w_{ij})$$

$$\mathbf{\Psi p} \;=\; \begin{bmatrix} \phi_1 & 0 & & 0 \\ \phi_2 & 0 & & 0 \\ \vdots & \vdots & & \vdots \\ \phi_N & 0 & & 0 \\ 0 & \phi_1 & & 0 \\ 0 & \phi_2 & & 0 \\ \vdots & \vdots & & \vdots \\ 0 & \phi_N & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & & \phi_1 \\ 0 & 0 & & \phi_2 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & & \phi_N \end{bmatrix}^T \begin{bmatrix} w_{11} \\ w_{12} \\ \vdots \\ w_{1N} \\ w_{21} \\ w_{22} \\ \vdots \\ w_{2N} \\ \vdots \\ w_{M1} \\ w_{M2} \\ \vdots \\ w_{MN} \end{bmatrix}$$

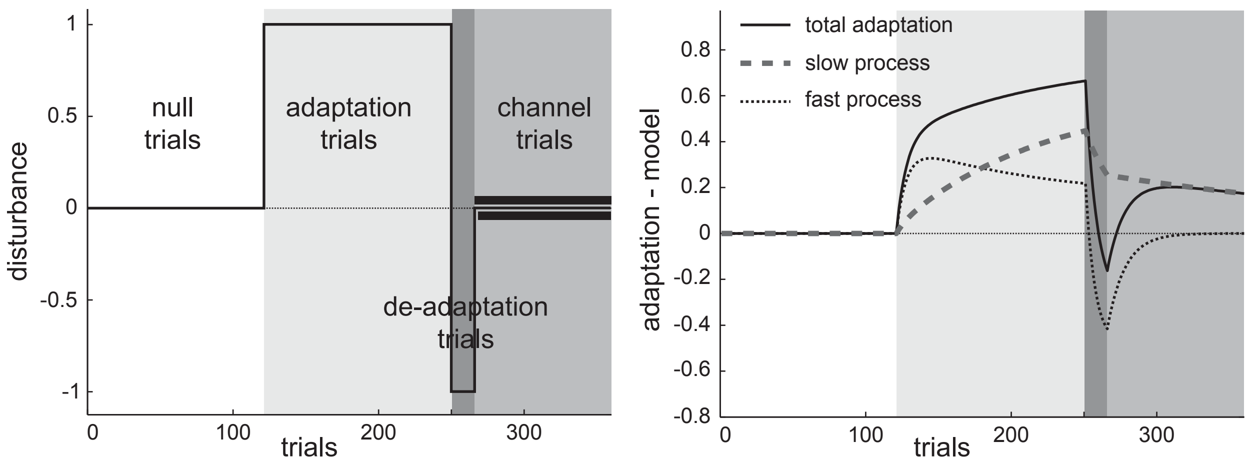
adaptation law $\mathbf{p}^{k+1} = \mathbf{p}^k + \alpha \mathbf{\Psi}^T \boldsymbol{\tau}_{FB}$ yields

$$\mathbf{W}^{k+1} = \mathbf{W}^k + \triangle \mathbf{W}^k, \;\; \triangle w_{ij}^k = \alpha\, v_i\, \phi_j\,, \;\; \alpha > 0$$

MOTOR MEMORY (1)

- the acquired internal models persists after training stops
- there is evidence of **saving**: faster re-adaptation to the same force field
- there is also **interference** between force fields learned consecutively
- colour context does not seem to help differentiating force fields to learn, but associated movement can enable learning several force fields simultaneously

MOTOR MEMORY (2)

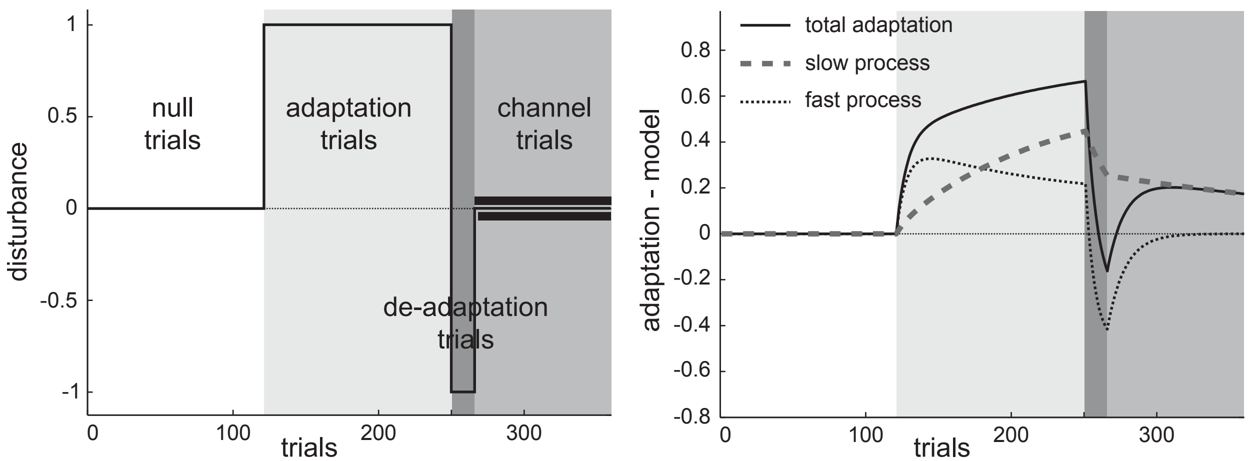


adaptation is formed of two processes with different time constants:

$$
\begin{aligned}
x^i &\equiv x_f^i + x_f^i \\
x_f^{i+1} &= \eta_f x_f^i + \alpha_f e^i \\
x_s^{i+1} &= \eta_s x_s^i + \alpha_s e^i
\end{aligned}
$$

x_f: fast process, x_s slow process
$0 < \eta_f < 1$, $0 < \eta_s < 1$: retention factors
$0 < \alpha_f$, $0 < \alpha_s$ learning rates

MOTOR MEMORY (3)



- the fast process may represent error-dependent feedback correction and occur in the cerebellum

- the slow process may represent formation of predictive feedforward command which occurs in the primary motor cortex

SUMMARY

- the task dynamics of a human or robot arm can be learned using techniques from adaptive control theory and neural networks, which are all based on the gradient descent feedback error minimisation

- the adaptation law $\mathbf{p}^{k+1} = \mathbf{p}^k + \alpha \boldsymbol{\Psi}^T \boldsymbol{\tau}_{FB}$ can be applied to:
  - *iterative control*, to learn torque $\boldsymbol{\tau}^k$ along a repeated trajectory,
  - *nonlinear adaptive control*, to learn a parametric physical model $\boldsymbol{\Psi}\mathbf{p}$, and
  - an (e.g. radial basis) *artificial neural network* $\mathbf{W}\boldsymbol{\Phi}$ to learn unstructured dynamics

- motor memory depends on context, is characterised by the variables of the force fields, occurs through a fast and slow processes