Combining Modalities with Different Latencies for Optimal Motor Control

Fredrik Bissmarck[1,2,3], Hiroyuki Nakahara[4], Kenji Doya[1,5], and Okihide Hikosaka[6]

Abstract

■ Feedback signals may be of different modality, latency, and accuracy. To learn and control motor tasks, the feedback available may be redundant, and it would not be necessary to rely on every accessible feedback loop. Which feedback loops should then be utilized? In this article, we propose that the latency is a critical factor to determine which signals will be influential at different learning stages. We use a computational framework to study the role of feedback modules with different latencies in optimal motor control. Instead of explicit gating between modules, the reinforcement learning algorithm learns to rely on the more useful module. We tested our paradigm for two different implementations, which confirmed our hypothesis. In the first, we examined how feedback latency affects the competitiveness of two identical modules. In the second, we examined an example of visuomotor sequence learning, where a plastic, faster somatosensory module interacts with a preacquired, slower visual module. We found that the overall performance depended on the latency of the faster module alone, whereas the relative latency determines the independence of the faster from the slower. In the second implementation, the somatosensory module with shorter latency overtook the slower visual module, and realized better overall performance. The visual module played different roles in early and late learning. First, it worked as a guide for the exploration of the somatosensory module. Then, when learning had converged, it contributed to robustness against system noise and external perturbations. Overall, these results demonstrate that our framework successfully learns to utilize the most useful available feedback for optimal control. ■

INTRODUCTION

For motor control and learning, the brain relies on feedback signals of different modalities such as vision and somatosensation, and appears to use them selectively depending on the task demands and the extent of learning. For example, to play the piano, the novice must first rely on visual and somatosensory feedback for finger movement. With practice, she can gradually reduce the reliance of visual feedback, and, as expert, she does not need to look at the keyboard at all. How does the reliance of feedback change with learning? In this article, we consider a hypothesis that feedback delay is a major factor in selection of feedback modality, and test if appropriate feedback pathways can be selected through reinforcement learning to achieve the best real-time motor performance.

Recently, the optimal feedback control paradigm (Kording & Wolpert, 2006; Todorov & Jordan, 2002) has successfully predicted human motor behavior (Liu & Todorov, 2007; Kording & Wolpert, 2004; Todorov & Jordan, 2002). Under this framework, execution is preceded by state estimation, by integration of available feedback modalities and models. Here the current state would have to be inferred from the delayed feedback in a recursive manner. In the context of well-learned, specialized motor skills, which are characterized by fast execution and minimum effort, this may be computationally expensive and time-consuming. In the present study, we propose an alternative, model-free architecture for learning and control of motor skills, where motor commands are computed in parallel by a modular circuit for each modality. Motor commands are mapped directly to the crude, delayed sensory feedback, and then integrated with the outputs of other modalities. This way, the quickest feedback is directly available to the controller for exploitation.

The actor–critic (Barto, 1995) is a reinforcement learning architecture proposed to be implemented by the basal ganglia–thalamocortical (BG–TC) system (Doya, 1999; Montague, Dayan, & Sejnowski, 1996; Houk, Adams, & Barto, 1995). Our framework is an actor–critic architecture with multiple actors (Nakahara, Doya, & Hikosaka, 2001), where each actor corresponds to a modality or submodality. We propose that the feedback latency constrains the utility of each actor. Further, we propose that the reinforcement learning algorithm plays

[1]ATR International, Kyoto, Japan, [2]National Institute of Communication Technology (NICT), Kyoto, Japan, [3]NAIST, Nara, Japan, [4]RIKEN Brain Science Institute, Saitama, Japan, [5]Okinawa Institute of Science and Technology, Okinawa, Japan, [6]National Institutes of Health, Bethesda, MD

a critical role in gating between inputs—only the better feedback signals, presumably those with shorter latency, would be reinforced. Thus, modular inputs are, once learned, gated implicitly by their latency. In our framework, the gating is realized by a combination of population coded outputs, sharpened by a softmax function in favor of the module with highest confidence. This mechanism is different from explicit gating (Haruno, Wolpert, & Kawato, 2001; Jacobs, Jordan, & Barto, 1991), where explicit signals are computed to weight the influence of modules on the combined output.

The article is outlined as follows: first, we present the general framework of our model (General Framework section). Then, we outline two implementations for validation of our model—Experiment 1 and Experiment 2 (Visuomotor Reaching Tasks), with results of simulations presented in the following section (Simulation Results). In Experiment 1, we studied a very simple system to clearly understand the effect of feedback delays. Two modules, which are identical except for their feedback delays, were trained until convergence for a simple arm reaching task. We found that (1) performance was constrained by the latency of the faster module, and (2) that the contribution of a module depended on its latency relative to the other module. The faster module could effectively learn the task, without interference from the slower module. In Experiment 2, we studied the interaction between vision and somatosensation in a sequential reaching task. Here, we assumed that a ''somatosensory module,'' corresponding to a motor skill, is learned under the assistance of a ''visual module,'' a preacquired, general but suboptimal controller guided by visual feedback. We found that for the somatosensory module to become independent of the visual module, it is critical that the latency of the somatosensory module has the shorter latency. During learning, albeit its longer latency, the visual module still functioned as a teacher for the somatosensory module. Once the motor skill was sufficiently acquired by the somatosensory module, it could execute the movement alone, independent from the visual module. At this mature stage of learning, the visual module still contributed by maintaining robustness against unexpected perturbations. Given these results, we propose that our framework, combining reinforcement learning modules by a softmax combination of population codes, realizes flexible learning and robust motor control by utilizing the best available feedback. Its implications are presented in the Discussion section.

GENERAL FRAMEWORK

As a simple model of learning control using multiple delayed feedback channels, we consider a modular architecture as shown in Figure 1. The state $\mathbf{x}(t)$ of the physical environment evolves depending on the motor command $\mathbf{u}(t)$. The state is monitored through different sensory channels $\mathbf{y}^m(t)$ with different delays $\boldsymbol{\tau}^m$ ($m = 1, \ldots, M$). Each module outputs a population-coded motor command $\mathbf{a}^m(t)$, and through their combination $\boldsymbol{\pi}(t)$, the final motor command $\mathbf{u}(t)$ is sent out to the physical environment. The goal of control is to maximize the cumulative reward $r(t)$, as in the standard reinforcement learning paradigm (Doya, 2000; Barto, 1995).

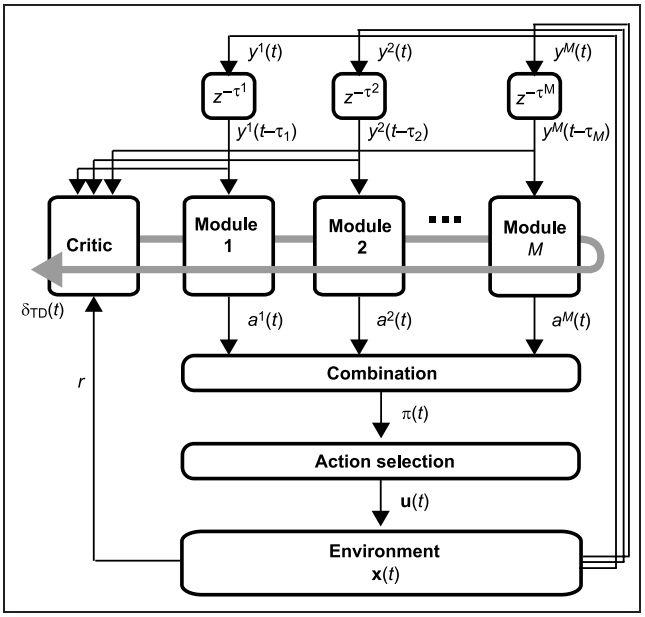

Figure 1. The modular network architecture for learning control with multiple feedback pathways (see text).

Below we outline the operation of the feedback control modules, combination of their outputs, and the learning algorithm. The architecture presented here is a modification of an earlier draft (Bissmarck, Nakahara, Doya, & Hikosaka, 2005).

Feedback Control Modules

Each module m has a characteristic feedback signal

$$\mathbf{y}^m(t) = \mathrm{f}^m(\mathbf{x}(t - \boldsymbol{\tau}^m)) \quad (1)$$

where $\mathrm{f}^m()$ is an observation function and $\boldsymbol{\tau}^m$ is a characteristic latency for the particular module. Each module gives as output a population code

$$\mathbf{a}^m(t) = \mathrm{g}(\mathbf{y}^m(t); \mathbf{w}^m) \quad (2)$$

where $\mathrm{g}(t)$ is a function approximator, with a set of trainable parameters $\mathbf{w}^m$. Each element $a_j^m(t)$ ($j = 1, 2, \ldots J$) corresponds to a preferred motor output $\bar{\mathbf{u}}_j$.

Combination of Modular Outputs

The motor command $\mathbf{u} \in R^D$ is represented by a combination of the population coded outputs of all modules with a softmax function:

$$\pi_j(t) = \frac{\exp\left(\beta \sum_{m=1}^{M} a_j^m(t) + n_j(t)\right)}{\sum_{j=1}^{J} \exp\left(\beta \sum_{m=1}^{M} a_j^m(t) + n_j(t)\right)} \quad (3)$$

where β is a constant that regulated the overlap of population codes. The noise term $n_j(t)$ makes the policy stochastic, that is, it controls the exploration of the agent. The actual motor command $\mathbf{u}(t)$ is given by the weighted sum of the preferred motor commands $\overline{\mathbf{u}}_j$ corresponding to each population code:

$$\mathbf{u}(t) = \sum_{j=1}^{J} \pi_j(t) \overline{\mathbf{u}}_j. \quad (4)$$

The modular outputs a_j^m can be interpreted as the log-probability of selecting the output $\overline{\mathbf{u}}_j$,

$$a_j^m(t) = \log(P(\overline{\mathbf{u}}_j(t) \mid \mathbf{y}^m(t - \tau^m), \mathbf{w}^m)). \quad (5)$$

Summing over all modules, adding noise, and exponentiating give the full probability $P(\overline{\mathbf{u}}_j(t)) = \pi_j(t)$. This simple but straightforward interpretation gives a direct relationship between the activities of single neurons and distributional population codes (Pouget, Dayan, & Zemel, 2003; Weiss & Fleet, 2002).

Actor–Critic Learning

Our model implements a form of the continuous actor–critic (Doya, 2000). The goal of learning is to maximize the cumulative future rewards:

$$E\left[\int_0^\infty e^{-\frac{s}{\tau^{\mathrm{TD}}}} r(t+s)\mathrm{d}s\right] \quad (6)$$

where τ^{TD} determines how far into the future returns should be considered.

The role of the critic is to estimate the cumulative future reward from each state $\mathbf{x}(t)$ in the form of state value function:

$$V(\mathbf{x}(t)) = E\left[\int_0^\infty e^{-\frac{s}{\tau^{\mathrm{TD}}}} r(t+s)\mathrm{d}s\right] \quad (7)$$

for each state $\mathbf{x}(t)$. The critic learns to estimate the value function from available feedback:

$$V\left(\mathbf{y}^1(t), \mathbf{y}^2(t), \ldots, \mathbf{y}^M(t); \mathbf{w}^c\right) \quad (8)$$

where $\mathbf{w}^c$ is a set of trainable parameters.

Learning of the critic and the feedback control modules is based on the temporal difference (TD) error

$$\delta^{\mathrm{TD}}(t) = r(t) - \frac{1}{\tau^{\mathrm{TD}}} V(t) + \dot{V}(t), \quad (9)$$

which signals the deviation of reward prediction (see Appendix A for the update equations for the critic parameters $\mathbf{w}^c$ and the controller parameters $\mathbf{w}^m$).

VISUOMOTOR REACHING TASKS

We test the effects of different sensory feedback delays in two simulated experiments of arm reaching. In Experiment 1, we used two somatosensory feedback control modules with different delays for a simple reaching task. The aim is to see how the minimal delay affects the control performance and how relative feedback delay affects the selection of the modules by learning. In Experiment 2, we used both visual and somatosensory feedback modules for a sequential reaching task. The aim is to see whether and how transition from slow, task-independent visual control to fast, task-dependent somatosensory control happens under different feedback delays.

Figures 2 and 5 show the implementation of Experiments 1 and 2, respectively.

Reaching Tasks

We use a 2DOF arm, where each link is 0.3 m long, 0.1 m in diameter, and 1 kg (see Figure 2). The state is defined by its shoulder and elbow joint angles θ_1 and θ_2 and angular velocities $\dot{\theta}_1$ and $\dot{\theta}_2$. The Cartesian hand position is $\xi^{\mathrm{hand}}(\theta_1, \theta_2)$. The arm moves according to the motor command $\mathbf{u}(t) = (u_1(t), u_2(t))$. In Experiment 1, we assume the system noise proportional to the motor command so that each joint torque is given by:

$$u_d^{\mathrm{actual}}(t) = (1 + n_d(t))u_d(t) \qquad (d = 1, 2) \quad (10)$$

where $n_d(t)$ is white noise with unit variance and mean zero. In Experiment 2, we assumed the system noise to be zero.

In Experiment 1, the goal is to move the hand as quickly and accurately as possible to the target position T given

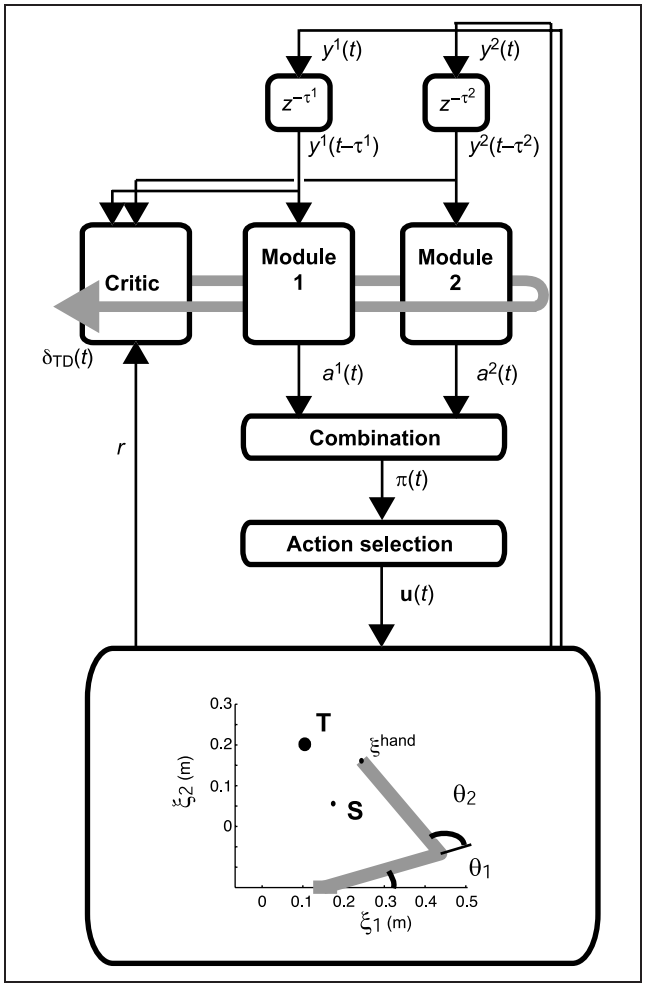

Figure 2. The implementation of (τ^1, τ^2) = Experiment 1. The agent controls a 2DOF arm by applying joint torques to shoulder and elbow joints, with angles θ_1 and θ_2, respectively (ξ_1 and ξ_2 define the Cartesian coordinates). The task is to reach from the start hand position S to the target position T, as quickly and accurately as possible (according to the reward signal r). The feedback signals $\mathbf{y}^1$ and $\mathbf{y}^2$ are identical, a population code of joint angles and velocities. See text for further details.

the start position S. The reward signal is given by an exponential function of the distance of the hand to the target

$$r(t) = a \exp\left(-b \parallel \xi^{\text{hand}}(t) - \xi^{\text{target}} \parallel\right) + c \qquad (11)$$

where $a = 6$, $b = 20$ and $c = -0.3$. Each trial lasts for 1.0 sec.

In Experiment 2, the task is to press three targets in consecutive order, which always appear one at the time at the same positions, marked 1, 2, and 3 in Figure 5. A target is pressed when the hand reaches a proximity of the target $\|\xi^{\text{hand}}(t) - \xi^{\text{target}}\| < \xi^{\text{prox}}$ at a low speed ($\|\dot{\xi}^{\text{hand}}(t) \| < v^{\text{prox}}$) ($\xi^{\text{prox}} = 0.02$ m and $v^{\text{prox}} = 0.5$ m/sec). After each successful target reaching, the agent is rewarded with an increasing amount (50, 100, and 150) and the next target appears immediately. Each trial ends after successful completion of the sequence, or after 5 sec.

Feedback Control Modules

In Experiment 1, we use two somatosensory feedback controllers, whereas in Experiment 2, we use somatosensory and visual feedback controllers.

The Somatosensory Module

The somatosensory control module uses a population code representing joint angles $\boldsymbol{\theta}$ and angular velocities $\dot{\boldsymbol{\theta}}$ of the arm as the input:

$$y_k^m(t) = \frac{1}{Z}\exp\left(-\frac{1}{2}\left\{ \sum_d \left(\frac{\theta_d(t-\tau_m) - \overline{\theta}_{kd}}{\sigma_{kd}} \right)^2 + \sum_d \left(\frac{\dot{\theta}_d(t-\tau_m) - \overline{\omega}_{id}}{\sigma'_{kd}} \right)^2 \right\} \right) \qquad (12)$$

where $k = 1,2,..K$ is the index of the input units, $\overline{\theta}_{kd}$ and $\overline{\omega}_{kd}$ are their preferred joint angles and velocities, σ and σ' are their width parameters, and Z is a normalization term. In Experiment 2, we introduced additional units representing the time since the target onset.

The output of module m is given by another population code

$$\mathbf{a}^m(t) = \mathbf{W}^m \mathbf{y}^m(t - \tau^m) \qquad (13)$$

where $\mathbf{W}^m$ are trainable weight matrices. Initially, all weights are zero (see Appendix B for more details).

The Visual Module

The input for the visual feedback controller is the Cartesian positions of the hand and the target

$$\mathbf{y}^v(t) = \left\{ \tilde{\xi}^{\text{hand}}(t), \xi^{\text{target}}(t - \tau^v) \right\}. \qquad (14)$$

Although the target position is subject to feedback delay τ^v, we assume that an estimate of the present hand position $\tilde{\xi}^{\text{hand}}$ is available, for example, by simple linear prediction. The output is expressed as a population code $\mathbf{a}^v$.

We assume that the feedback control of the visual module (indexed by v) is preacquired and use a linear

feedback controller with inverse dynamics compensation and output smoothing (see Appendix C). The controller produces a bell-shaped velocity profile similar to natural hand movement.

Actor–Critic Learning

To promote effective exploration, we use a low-pass filtered noise $\mathbf{n}$ in the action output (Equation 3)

$$\tau^n \dot{\mathbf{n}}(t) = -\mathbf{n}(t) + \nu \mathbf{N}(t) \tag{15}$$

where the time constant $\tau^n = 50$ msec and $\mathbf{N}(t)$ is Gaussian noise with zero mean and unit variance. The amplitude ν is fixed at 0.1 in Experiment 1, and reduced as $1/(1 + 0.0001T)$ at trial T in Experiment 2.

The critic takes the population-coded somatosensory feedback as its input and uses a linear weighting to produce the value estimate

$$V(t) = \mathbf{w}^c \mathbf{y}(t), \tag{16}$$

where $\mathbf{y}(t) = (\mathbf{y}^1(t), \mathbf{y}^2(t))$ in Experiment 1 and $\mathbf{y}(t) = \mathbf{y}^s(t)$ in Experiment 2. (We verified that inclusion of visual input $\mathbf{y}^v$ in Experiment 2 did not affect the results in this example.)

Performance Measures

In order to compare the control performance under different settings of feedback delays, we use a number of performance measures, namely, the hand trajectory, the hand velocity profile, the cumulative reward, and the performance time.

The cumulative reward is given by

$$R = \int_0^T r(t) \mathrm{d}t, \tag{17}$$

where T is the length of a trial ($T = 1$ sec in Experiment 1, $T = 5$ sec in Experiment 2).

To compare the relative contribution of different modules, we define the actor weight ratio, the output deviation, and the relative output proximity. The actor weight ratio (AWR) is defined as the ratio of the absolute sum of actor weights of respective trained module:

$$\mathrm{AWR} = \frac{\sum_k \sum_j | w_{jk}^1 |}{\sum_k \sum_j | w_{jk}^2 |}, \tag{18}$$

that is, a value AWR > 1 indicates a relatively more influential actor of Module 1. We also define the output deviation

$$d^m(\mathbf{x}(t)) = \mathbf{u}(\mathbf{x}(t)) - \mathbf{u}^m(\mathbf{x}(t)) (m = v, s) \tag{19}$$

which shows how much module m's output differ from the agent's at time t. Its time average over trajectories is given by

$$\langle d^m \rangle = \frac{1}{T} \int_0^T | d^m(t) | \mathrm{d}t. \tag{20}$$

for a trial terminating at T. From the output deviation, we define the relative output proximity

$$p^v(t) = 1 - \frac{d^v(t)}{d^v(t) + d^s(t)} \tag{21}$$

$$p^s(t) = 1 - p^v(t) \tag{22}$$

which is a measure of the relative influence of visual and somatosensory modules over agent output, respectively. Both $p^v(t)$ and $p^s(t)$ are bounded between 0 and 1 and their relation is $p^v(t) + p^s(t) = 1$ by definition. Thus, the value of one module larger than that of the other module indicates that the former is a dominant module.

Simulation

In both experiments, the same learning parameters were used: inverse temperature $\beta = 10$, time constants $\tau^{\mathrm{TD}} = 200$ msec, $\tau^{\mathrm{ET}} = 200$ msec, and learning rate $\alpha = 0.1$ sec^{-1}.

All differential equations were approximated with the Euler forward method with a time step small enough not to affect the results (10 msec).

SIMULATION RESULTS

Experiment 1: Simple Reaching with Somatosensory Feedback Modules with Different Delays

In Experiment 1 (Figure 2), we investigate how feedback latencies affect learning and control in the proposed framework. We use a simple implementation with two identical modules, except for their feedback latencies. The task is to learn a simple reaching movement with a 2DOF arm. We train the networks with different pairs of feedback latencies and compare their performance and relative contribution of modules after learning has converged.

Reaching Trajectories

Figure 3 shows examples of hand trajectories generated by the architecture with four different settings of feedback latencies. The 10 trajectories in the top row (Figure 3A) are generated with both modules after 100,000 training trials. Effective reaching movement is achieved by all four latency pairs in a robust manner. In the case of $(\tau^1, \tau^2) = (0, 50)$, the variability is higher than in the other examples. Because the reward function (see Reaching Tasks) does not explicitly penalize variability, this is still a performance optimally close to other well-performing agents (see below). However, in the cases of $(\tau^1, \tau^2) = (0, 0)$, and (0, 50) msec, the movement is much faster than (50, 50), and (50, 100) msec, as can be seen in the hand velocity plots in the bottom row (Figure 3D, solid lines; mean velocity of the samples in Figure 3A). This shows that the shortest feedback delay is critical for the performance. This is not a trivial finding as the output of the module with the longer feedback delay can interfere with the feedback command generated by the module with shorter delay.

In order to see the relative contribution of the two modules, we compared the trajectories generated by either one of the modules (Figure 3B and C) with the other module's output set as $a^m(t) = 0$. With the identical delays $(\tau^1, \tau^2) = (0, 0)$ and (50, 50) msec, both module can realize comparable trajectories. On the other hand, with different delays $(\tau^1, \tau^2) = (0, 50)$ and (50, 100) msec, although Module 1 with shorter delay can realize nice trajectories, Module 2 with longer delays generates very poor trajectories. This shows that the less desired outputs of the module with longer feedback delay is effectively shut down by reinforcement learning.

Effects of Minimum and Relative Delays

To verify the critical role of the minimum feedback delay in control performance and the role of relative feedback delay for module selection, we measured the cumulative reward R and the actor weight ratio of trained agents for 24 different pairs of feedback delays. Under the condition of $\tau^1 \leq \tau^2$, we plot those measures in the parameter space of $\tau^{min} = \tau^1$ and $\Delta\tau = \tau^2 - \tau^1$.

Figure 4A shows how the cumulative reward depends on τ^{min} and $\Delta\tau$. Each black dot corresponds to a trained agent with the specific latency pair. It is clearly seen that the longer τ^{min} results in reduced cumulative reward, whereas the relative delay $\Delta\tau$ has almost no effect on the performance. Figure 4B shows the actor weight ratio, which increases markedly with the increase of the relative delay $\Delta\tau$ (for $\tau^{min} = 0$, per definition AWR $\equiv$ 1. Never was AWR $<$ 1.).

These results confirm that the performance of the modular learning control architecture is mostly determined by the module with shortest latency. This is achieved by the softmax combination of population-

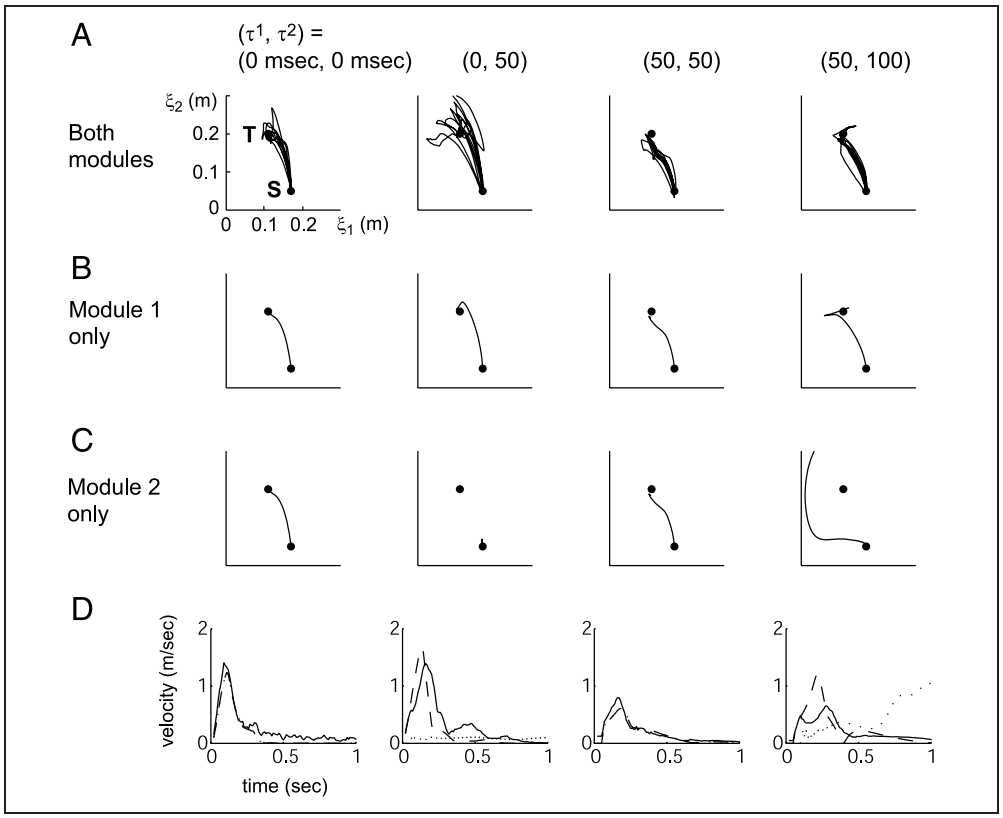

Figure 3. Trajectory samples generated by four different settings of latencies $(\tau^1, \tau^2) =$ (0, 0), (0, 50), (50, 50), and (50–100) msec after 100,000 trials of training. S = start; T = target. (A) Ten trajectory samples generated by both modules under system noise. (B) System noise-free trajectory generated by Module 1 only. (C) System noise-free trajectory generated by Module 2 only. (D) Velocities of both modules (solid line, mean of 10 samples), Module 1 only (dashed line) and Module 2 (dotted line).

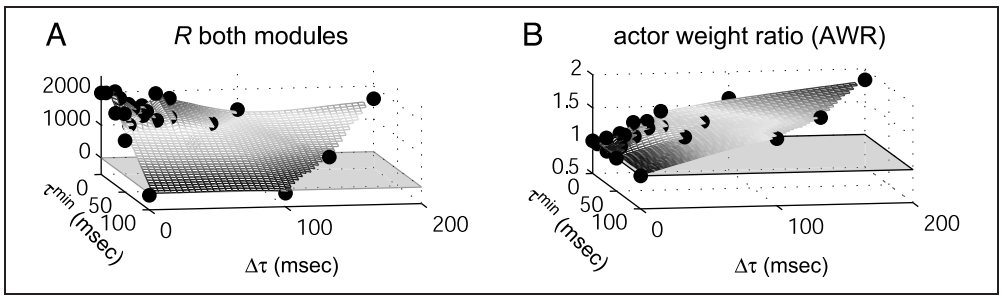

Figure 4. Latency-dependent performance measures, displayed as surface plots constrained by 24 latency pairs (black dots) over latency space (τ^{min}, $\Delta\tau$). (A) Cumulative reward *R*, indicating behavioral performance of agent. (B) Actor weight ratio (AWR), indicating relative contribution of modules. A value below the *x*–*y* plane (AWR = 1) indicates relatively larger contribution of slower Module 2, above the plane indicates larger contribution of faster Module 1.

coded outputs of modules (see Combination of Modular Outputs) and tuning of modular outputs by actor–critic learning. It is noteworthy that potential problem of slower feedback module contaminating the good output of the faster module has been avoided by this scheme.

Experiment 2: Sequential Reaching with Visual and Somatosensory Feedbacks

In Experiment 2 (Figure 5), we introduce a more realistic, complex implementation of a visuomotor sequence task. In motor skill acquisition, there is substantial evidence for a shift in cortical activity with experience, from prefrontal areas to motor areas (Floyer-Lea & Matthews, 2005; Hikosaka, Nakamura, Sakai, & Nakahara, 2002; Petersen, van Mier, Fiez, & Raichle, 1998; Jueptner, Frith, Brooks, Frackowiak, & Passingham, 1997; Doyon, Owen, Petrides, Sziklas, & Evans, 1996). Analogously, there should be a shift in modalities of feedback subserving these cortical areas; from extrinsic (visual) feedback needed for anticipation and proceduralization of task dynamics to intrinsic (somatosensory) feedback needed for optimization of motor control (Nakahara et al., 2001).

Here, we study the transfer between these two systems, a ''visual module'' and a ''somatosensory module,'' in a task of reaching a stereotyped sequence of three targets. The visual module relies on a general purpose controller which regulates a single reach to a given visual target. We assume that the module is preacquired and is not to be optimized for any particular target sequence. The somatosensory module relies on somatosensory feedback and becomes optimized for repeated motor sequences. Architectures with different latency pairs τ^v and τ^s were trained for 100,000 trials, after which learning had converged in all cases. We investigate the relative contribution of the somatosensory module for different latency pairs and also compare the robustness against external perturbations of the composite system versus single module control.

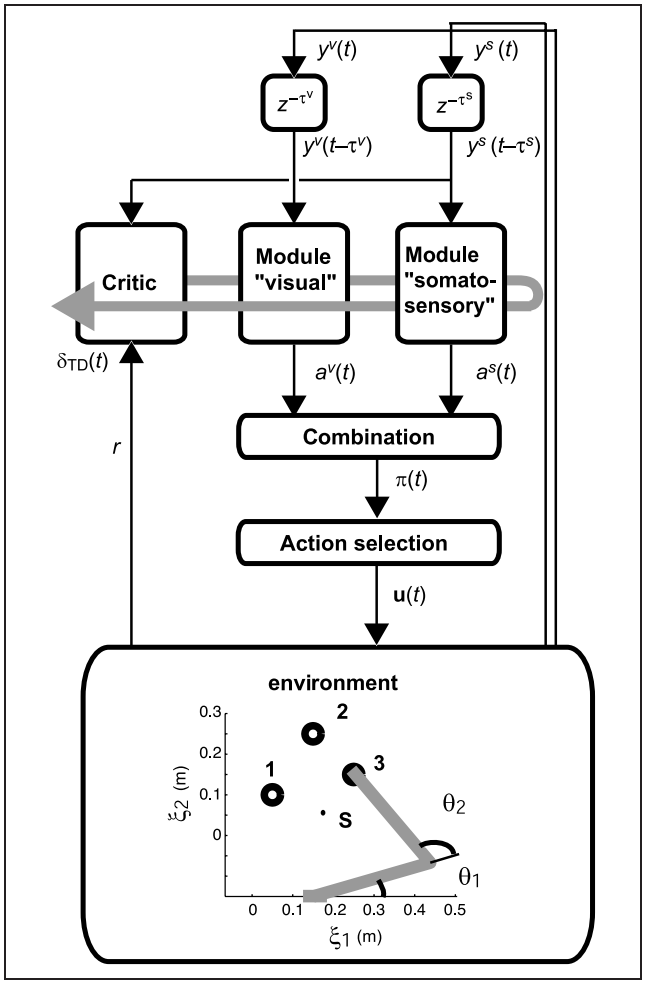

Figure 5. The implementation of Experiment 2. Here, with the arm as in Experiment 1, the goal is to press Targets 1, 2, and 3, presented in consequent order, starting from S. Reward is given only at the time when a key is pressed. The agent consists of two modules called ''visual module'' and ''somatosensory module.'' The visual module is a fixed controller, receiving feedback about the current target position ξ^{target} and hand position ξ^{hand} to control a reaching movement. The somatosensory module is similar to the modules in Experiment 1. See text for further details.

Learning Performance

Figure 6A and B compares the reaching trajectories before and after learning [(τ^v, τ^s) = (100, 0) msec]. Before learning, movements are variable and step-by-step—they are directed toward one target at the time. After learning, movements are stereotyped, and also coarticulated, as they are redirected toward Targets 2 and 3 before preceding targets are concluded.

Figure 6C compares the performance time (the time it takes to complete one trial) before (black bars) and after (white bars) 100,000 trials of learning for 12 different latency pairs. Clearly, sequence-specific learning by the somatosensory module contributes to reduction of the performance time. Its potential to do so is primarily constrained by τ^s, which has a decreasing trend of performance time for 100, 50, and 0 msec with any latency τ^v of the visual module. In turn, τ^v is also a constraint for performance, as the performance times of learned modules are shorter with lower τ^v. The overall similar variance of performance time suggests that, within the range of investigated delays, robustness to noise is not affected by delay in our composite system.

Contribution of the Somatosensory Module

To elucidate the contribution of the somatosensory module, we compared the joint torque outputs of single modules (computed as in Experiment 1) with the joint torque output of the agent. Figure 7A and B shows trajectories of generated joint torques over time (one trial) for the two extremes of relative latency in our study: [(τ^v, τ^s) = (100, 0) msec (A), $\Delta\tau$ = 100 msec] and [(τ^v, τ^s) = (0, 100) msec (B), $\Delta\tau$ = −100 msec]. In the first latency pair, the somatosensory module generates an output different from the visual module, but is evidently dominant as it is close to the agent output. In the second latency pair, the outputs of the two modules are close to each other, indicating that both modules equally contribute to the agent output. Figure 7C shows the quantitative picture, expressed as mean output deviation for seven latency pairs. In cases of $\tau^v < \tau^s$, the visual module has the smaller output deviation [(50, 100) and (0, 100)], indicating a larger contribution. In the case of mutual, long latency (100, 100), contribution is equal. Otherwise, the somatosensory module has lower output deviation. This result indicates that for the somatosensory module to learn an independent policy, it needs to have a shorter or equal latency τ^s relative to τ^v, that is, $\tau^s \leq \tau^v$.

We then investigated how the learned behavior is driven by the somatosensory module. We compared the normal behavior of the learned agent with a condition with the visual module inactive. Figure 8A shows examples of hand trajectories for four latency pairs in the two conditions. With both modules, all agents are always successful. When the visual module is inactive, the ability to control the movement depends on the

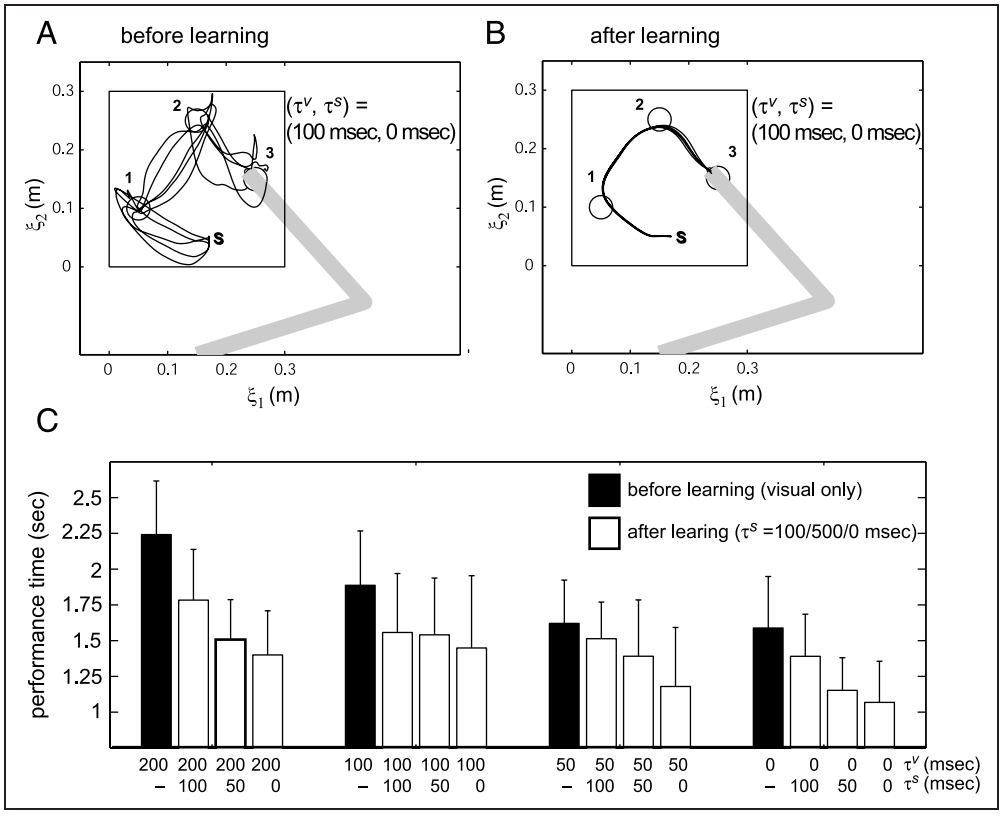

Figure 6. Performance before and after learning. (A) Five sample trajectories before learning. (B) Five sample trajectories after learning. (C) Performance times of 12 latency pairs before learning (black bars) and after learning (white bars), compared with equal levels of exploratory noise (ν = 0.01). Note that the initial performance is the same for agents with equal τ^v, as the somatosensory module is inactive before learning.

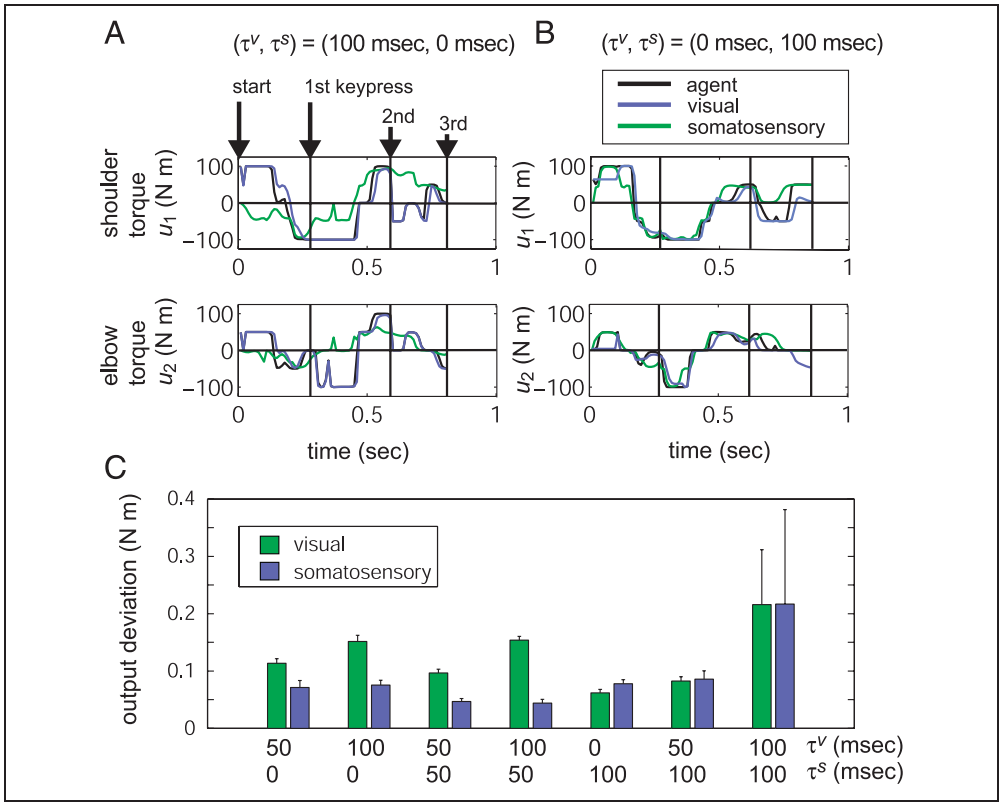

Figure 7. A comparison of contribution to joint torque outputs between the visual and somatosensory modules. (A–B) Example trajectories of shoulder (top) and elbow (bottom) torques over time for the latency pairs (τ^V, τ^S) = (100, 0) (A) and (0, 100) msec (B). The green, blue, and black lines correspond to the outputs of the visual module, somatosensory module, and agent, respectively. (C) A comparison of mean output deviation (100 trials, noise amplitude $\nu = 0.02$) of visual and somatosensory modules for seven latency pairs.

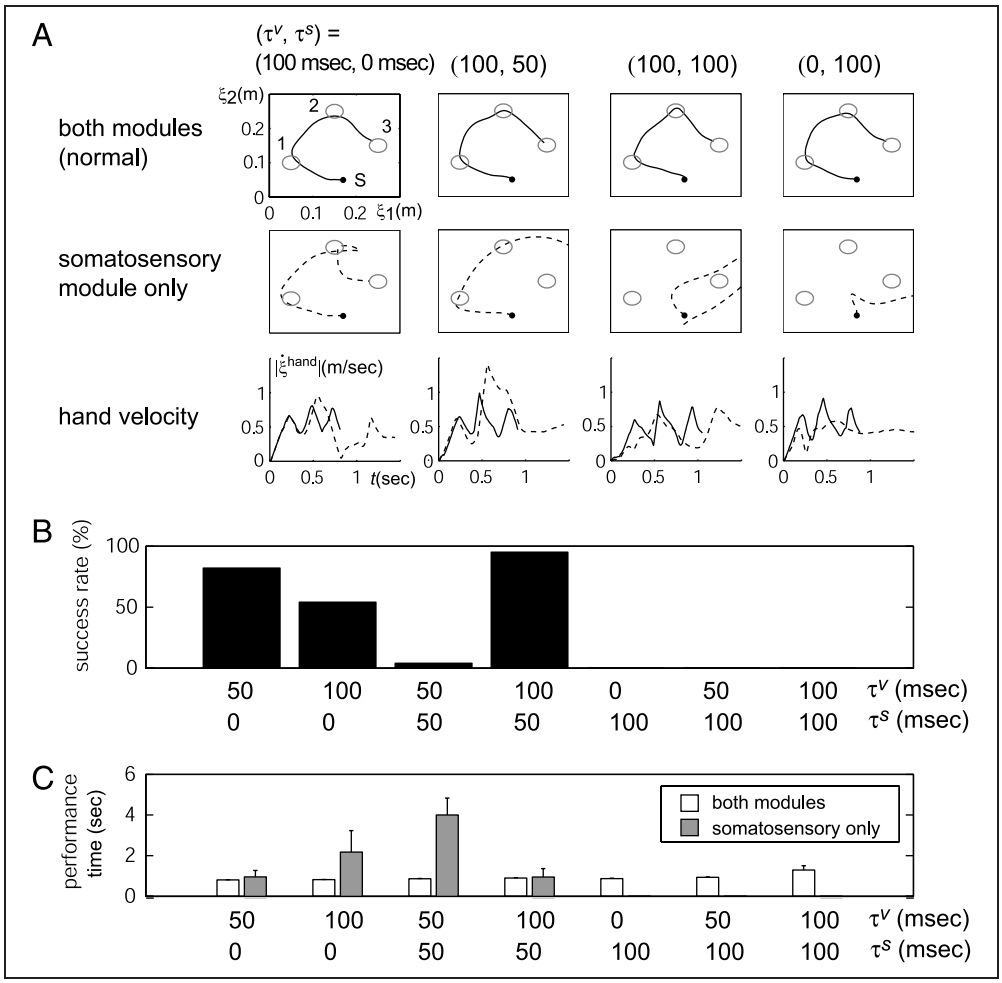

Figure 8. (A) Learned behavior of the agent in normal execution ("both modules"; top row, solid lines) and in execution with the visual module inactive ("somatosensory module only"; middle row, dashed lines) for four different latency pairs (noise-free). The bottom row shows corresponding, absolute hand velocities (first 1.5 sec) over time. (B) Success rates of the "somatosensory module only" condition for seven different latency pairs ($\nu = 0.02$). For the agents of $\tau^S = 100$, there were no successful trials in this condition. (C) Comparison of performance times between normal (white bars) and somatosensory module only (gray bars) conditions, for successful trials.

relative latency $\Delta\tau = \tau^v - \tau^s$. The success rate of the somatosensory module to complete a trial (given 100 trials) is shown in Figure 8B. We observe that the successful rate is high in the case of $\tau^s < \tau^v$, whereas none [or single trials in the case of (50, 50)] was successful in the case of $\tau^s \geq \tau^v$. These results further confirm our observation above (Figure 7) that the somatosensory module can become dominant as far as $\tau^s \leq \tau^v$.

Figure 8C compares the mean performance time of the two conditions. Two of the agents [(50, 0) and (100, 50)] can, on average, perform almost as well in the somatosensory-only condition, but note the smaller variance of the normal condition. The visual module provides robustness also late in learning.

Robustness to Perturbation

To further evaluate the robustness of the composite system, we perturbed a behaving agent [$(\tau^v, \tau^s) = (100, 0)$ msec] by applying a force on the end effector (hand). An impulse with constant force (400 N) was applied for 50 msec in a random direction (in the plane of the arm), for 50 msec, 0.3–0.6 sec after trial start. Figure 9A and B shows an example trajectory, where the impulse (400 N up left, onset at 0.3 sec) throws the agent off track to miss Target 2 to the left. The green/blue colors of the trajectory indicate the relative proximity (see Performance Measures) in Figure 9A of visual/somatosensory modules' output to the agent's, respectively. Note how the visual module predominates after the perturbation to put back the trained movement on track (toward Target 2), after which the somatosensory module regains influence anew. Figure 9C shows a comparison of the impact on performance time (mean of 1000 trials) for the random impulse, when operating with both modules or a single module active. The visual module functions as a safeguard against perturbations because the somatosensory module alone (blue bar) cannot effectively recover, resulting in the significantly higher performance time (for which 56% of the trials were timed out at 5.0 sec). The somatosensory module contributes to speedup before and after perturbation recovery, which is why both modules (black bar) are performing faster than the visual only (green bar).

In summary, these results indicate that, in this visuomotor sequence task, as learning progresses, the somatosensory module with the presumably shorter latency becomes dominant in motor control. After learning, the visual module provides stability when the effector ends up outside the well-trained regime. The memory transfer, or the degree of control by different modalities, critically depends on the difference in latencies between the visual and somatosensory modules.

DISCUSSION

We have examined how feedback latency affects the relative importance of modules for the learning and control of real-time motor skills. With softmax combination of population-coded output of multiple control modules, we demonstrated in simulations how the modules with shorter latency attain dominance in motor control. Although the result may sound straightforward, there are potential problems with conflicts of multiple modules, for instance, the longer latency output pulling back movement by the shorter latency module. It is

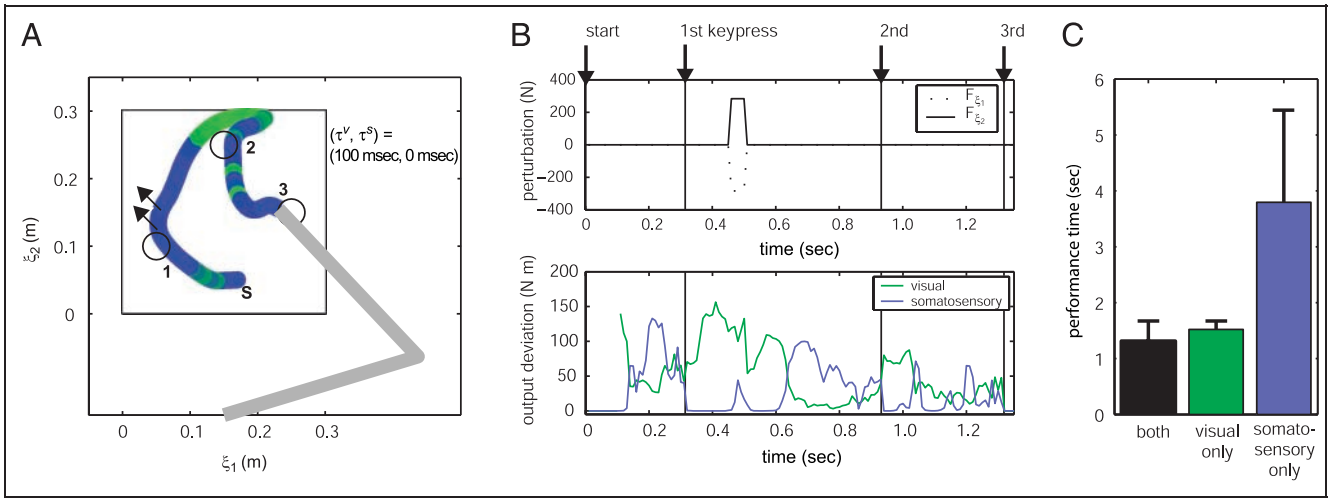

Figure 9. External force perturbation imposed on the end effector (hand) for a trained agent $(\tau^v, \tau^s) = (100, 0)$ msec. (A–B) Example trajectory when an impulse (400 N, 50 msec) perturbs the composite system. (A) Spatial movement trajectory. The two arrows indicate the direction of the force perturbation, drawn from the position of start and stop of the impulse. The green/blue color indicates relative proximity to visual/somatosensory modules' output, respectively, that is, green indicates $p^v(t) > p^s(t)$ and blue $p^v(t) < p^s(t)$. (B) Temporal trajectories (corresponding to A) of impulse (top), and output deviations of visual (green) and somatosensory (blue) modules. Note that output deviation is inverse proportional to proximity in (A). (C) Mean PT for perturbed trials with an impulse (400 N, 50 msec) of random direction and random onset (0.3–0.6 sec from trial start) for the agent $(\tau^v, \tau^s) = (100, 0)$ msec, comparing control with both versus single module.

noteworthy that appropriate module selection was achieved without any explicit gating and simply by reinforcement of the output of the module that best contributed to the performance.

The use of population codes strongly contribute to the robustness and flexibility of our framework. On the level of perception, population coding has been shown to help robustness against sensor noise (Pouget, Dayan, & Zemel, 2000; Georgopoulos, Kettner, & Schwartz, 1988). Our model shows the advantage of population coding at the level of action selection, with more effective weight on the module with a sharper output distribution. In Experiment 2, this explains the robustness of the composite system in the face of noise and force perturbations. The somatosensory module predominates the output within its experienced, expert regime, but as soon as noise or mechanical force makes the arm go out of range, the visual module resumes control.

In dealing with delayed or noisy sensory signal, a recently popular paradigm is to use recursive Bayesian filters to estimate the hidden state (Todorov, 2004). Such a model may also explain more weight on the faster module, which is more informative about the current state. However, Bayesian inference requires the models of the physical dynamics and sensory delay and noise, and also takes heavy on-line computation, except for linear Gaussian systems where Kalman filtering is possible. Instead, here we pursued a much simpler approach of training feedback controllers specialized for given delays. Analysis of pros and cons of these approaches and their possible integration is the subject of our future study.

The effects of feedback delays are a relatively less investigated aspect of motor control and learning (Miall & Jackson, 2006). Some tracking experiments have been conducted, all showing worse performance for artificially imposed delays 100 msec and longer (Ogawa, Inui, & Sugio, 2007; Miall & Jackson, 2006; Foulkes & Miall, 2000; Miall, Weir, & Stein, 1985). Kitazawa et al. also showed that learning speed and performance of prism adaptation of a reaching task worsened with delays of visual feedback of the end-point error for humans (Kitazawa, Kohno, & Uka, 1995) and monkeys (Kitazawa & Yin, 2002).

In Experiment 2, faster movements were learned even though reward was given only for keypresses, regardless of time expenditure. Rewards received faster are valued higher because of temporal discounting of rewards (Equation 6). This property may naturally explain why performance of numerous skill learning tasks (e.g., Anderson, 1995) speeds up, although speed is not an explicit performance criterion. Brain mechanisms of reward discounting is an active research topic (McClure, Ericson, Laibson, Loewenstein, & Cohen, 2007; Schweighofer et al., 2006; Tanaka et al., 2004; Daw & Touretzky, 2002).

The mechanism of transfer from declarative to procedural memories is poorly understood (Doyon & Benali, 2005; Hikosaka et al., 2002). In our framework, modules with shorter latency become dominant with learning. As demonstrated in Experiment 2, this allows specialized motor skills based on fast, intrinsic feedback loops to emerge under general purpose controllers based on slow, extrinsic feedback such as vision or audition. If the difference in feedback latency is long enough, the faster modality will eventually become independent of the slower modality, which can then be used for other purposes.

There are two analogies between our framework and the BG–TC system: (1) its organization into modular circuits (Alexander & Crutcher, 1990), and (2) the actor–critic architecture (Houk et al., 1995). In previous experimental (Hikosaka et al., 1999, 2002) and computational (Nakahara et al., 2001) work, we have proposed that prefrontal and motor BG–TC loops cooperate in motor sequence learning, encoding sequences in visual and motor coordinates, respectively.

The success of this rather simple modular learning control framework motivates future studies with agents comprising of more complex, heterogeneous features, such as different sensor noise levels, learning speeds, or inclusion of feedforward components. For example, given a slow, low-noise module and a fast, noisy module, the former would be used for precision tasks and the latter for speed tasks. To further test the generality of this prediction, delayed auditory feedback could be added as a third modality, and modality dependence could be tested under different pairs of feedback delays.

The brain receives possibly thousands of sensory signals from which it has to make a sensible response. Biological reinforcement learning may not just be about selecting actions, but also about selecting sensory input. In this context, feedback latencies may be a critical factor for which input and output connections are formed.

APPENDIX

A. Learning Algorithm

Our model implements a form of the continuous actor–critic (Doya, 2000). The function of the critic is to estimate the cumulative sum of expected future reward $r(t)$, that is, to learn the value function. For a given policy, the continuous value function is defined as

$$V(\mathbf{x}(t)) = E\left[\int_0^\infty e^{-\frac{s}{\tau^{\mathrm{TD}}}} r(t+s)\mathrm{d}s\right] \qquad (23)$$

for each state $\mathbf{x}(t)$. The time constant $\boldsymbol{\tau}^{\mathrm{TD}}$ determines how far into the future returns should be considered. The critic implements a function approximator to estimate the value function from available feedback:

$$V = V\left(\mathbf{y}^1\left(t-\boldsymbol{\tau}^1\right), \mathbf{y}^2\left(t-\boldsymbol{\tau}^2\right), \ldots, \mathbf{y}^M\left(t-\boldsymbol{\tau}^M\right); \mathbf{w}^c\right) \quad (24)$$

where $\mathbf{w}^c$ is a set of trainable parameters. The temporal difference (TD) error δ^{TD} is the discrepancy between expected and actual return $r(t)$. In its continuous form (Doya, 2000):

$$\delta^{\mathrm{TD}}(t) = r(t) - \frac{1}{\tau^{\mathrm{TD}}} V(t) + \dot{V}(t). \tag{25}$$

The TD error is used to update the parameters in the critic (see below), and converges to zero when Equation 8 is equal to Equation 6. The TD error is also used to improve the policy $\hat{\pi}(t)$ of the actor, where circumflex denotes the noise-free, deterministic policy. The action deviation signal

$$E_j(t) = \frac{(\pi_j(t) - \hat{\pi}_j(t))^2}{2} \tag{26}$$

is the difference between the learned action and the action that was actually selected. The TD error reinforces or penalizes this deviation to update the policy $\hat{\pi}$. To control the time scale of states and actions to be updated, we use eligibility traces:

$$\dot{e}^c_k(t) = -\frac{1}{\tau^{\mathrm{ET}}} e^c_k + \frac{\partial V}{\partial w^c_k} \qquad \dot{e}^m_{kj}(t) = -\frac{1}{\tau^{\mathrm{ET}}} e^c_{kj} + \frac{\partial E_j(t)}{\partial w^m_{kj}} \tag{27}$$

for the critic and actor modules, respectively. The parameters are indexed by k and τ^{ET} is a time constant. The trace for the mth actor is given from

$$\frac{\partial E_j(t)}{\partial w^m_{kj}} = (\pi_j(t) - \hat{\pi}_j(t)) \frac{\partial \pi_j(t)}{\partial w^m_{kj}}. \tag{28}$$

The parameters are updated by gradient descent as

$$\dot{w}^c_k = \alpha \delta^{\mathrm{TD}}(t) e^c_k(t) \qquad \dot{w}^m_{kj} = \alpha \delta^{\mathrm{TD}}(t) e^m_{kj}(t) \tag{29}$$

where α denotes the learning rate.

B. Population Codes

In both experiments, the population codes were equal. In the somatosensory modules, the preferred joint angles $\overline{\theta}_{kd}$ and angular velocities $\overline{\omega}_{kd}$ were distributed uniformly in a $7 \times 7 \times 3 \times 3$ grid ($K_0 = 441$ nodes) for $k = 1, 2, \ldots K_0$ nodes, in the ranges (-0.2:1.2, 1.2:1.6) rad and (-1:1, -1:1) rad/sec. The corresponding variances σ_{kd} and σ'_{kd} were half the distance to the closest node in each direction.

The preferred joint torques $\overline{\mathbf{u}}_j$ corresponding to action j were distributed symmetrically over the origin in a 5×5 grid, in the range (-100:100, -100:100) N m with the middle (0,0) unit removed. The corresponding variances σ''_{jd} were half the distance to the closest node in each direction.

The somatosensory module in Experiment 2 also included context units. The context units consists of three tapped delay lines, each corresponding to a key in the sequence task. Each delay line had 8 units (i.e., 24 context units in all). For the kth unit in the nth delay line ($k > K_0$, $k \neq K_0 + 8(n - 1) + 1$):

$$\dot{y}^m_k(t) = -\frac{1}{\tau^C} y^m_k(t) + y_{k-1}(t) \tag{30}$$

where $\tau^C = 30$ msec. Each delay line is initiated by the input at ($k = K_0 + 8(n - 1) + 1$):

$$y^m_k(t) = \delta\left(t - \tau^{\mathrm{keypress}}_n\right) \tag{31}$$

where δ is the Dirac delta function, and $\tau^{\mathrm{keypress}}_n$ is the instant the nth key was pressed.

C. The Visual Controller in Experiment 2

The feedback signal $\mathbf{y}^v$ to the visual module consists of the hand kinematics ξ^{hand}, $\dot{\xi}^{\mathrm{hand}}$ and the target position ξ^{target}. Because the computed torque control law itself does require at least a good estimation of the current motor kinematics, the delayed feedback signals will not produce satisfactory control: the delays will cause oscillations and will become unstable at some 50–100 msec. To overcome this problem, we assumed that the agent has a good model of its own internal dynamics and can cancel out the delay of ξ^{hand} with a prediction $\tilde{\xi}^{\mathrm{hand}}(t) = \xi^{\mathrm{hand}}(t)$. The target position ξ^{target} is assumed not to be predictable. Thus, with the onset of a new target, it takes τ^v msec before the visual module reacts toward that target. The control is further perturbed by a decoding error, by modification of the somatosensory module and by the stochasticity of action selection.

The joint torques are first computed by

$$\dot{\mathbf{u}}^{\mathrm{visual}}(t) = -\frac{1}{\tau^{CT}} \mathbf{u}^{\mathrm{visual}}(t) + \lambda \mathbf{u}^{\mathrm{visual}'}\left(\ddot{\tilde{\xi}}^{\mathrm{hand}}, \dot{\tilde{\xi}}^{\mathrm{hand}}, \mathbf{e}\right) \tag{32}$$

where τ^{CT} and λ are constants, $\mathbf{e} = \xi^{\text{target}}(t - \tau^{v}) - \xi^{\text{hand}}(t)$ and the input to the filter is the inverse dynamics equation

$$\mathbf{u}^{\text{visual}'}(t) = \mathbf{J}^{T}\left(\mathbf{M}\left(\ddot{\xi}^{\text{hand}} + \mathbf{K}_1\dot{\xi}^{\text{hand}} - \mathbf{K}_2\mathbf{e}\right) + \mathbf{C}\dot{\xi}^{\text{hand}}\right) \tag{33}$$

in Cartesian coordinates, where $\mathbf{J}$ is the Jacobian ($\partial\theta/\partial\tilde{\xi}^{\text{hand}}$), $\mathbf{M}$ the moment of inertia matrix, and $\mathbf{C}$ the Coriolis matrix. Using a filter by Equation 32, more bell-shaped velocity profiles of the hand, similar to biological motion, are generated, in contrast to using Equation 33 directly.

The module output is an expansion of the joint torque $\mathbf{u}^{\text{visual}}$ on a population vector

$$a_j^{v}(t) = \frac{1}{Z}\exp\left(-\frac{1}{2}\left\{\sum_d\left(\frac{u_d^{\text{visual}}(t) - \overline{u}_{jd}}{\sigma''_{jd}}\right)^2\right\}\right) \tag{34}$$

where Z is the normalization term, $\overline{u}_{jd}$ is a preferable joint torque for Cartesian dimension d for vector element j, and σ''_{jd} is the corresponding variance.

The parameters of Equations 32 and 33 were τ^{CT} = 50 msec, λ = 100, $\mathbf{K}_1$ = [10 0; 0 10], $\mathbf{K}_2$ = [50 0; 0 50].

Acknowledgments

F. B. thanks Mitsuo Kawato, Erhan Oztop, and Jun Morimoto for comments on an earlier draft. This research was funded by Core Research for Evolutional Science and Technology (CREST), Japan Science and Technology Agency.

Reprint requests should be sent to Fredrik Bissmarck, ATR Computational Neuroscience Labs, 2-2-2 Hikaridai Keihanna Science City, Seika, Soraku, Kyoto 619-0288, Japan, or via e-mail: fredrik.bissmarck@gmail.com.

REFERENCES

Alexander, G. E., & Crutcher, M. D. (1990). Functional architecture of basal ganglia circuits: Neural substrates of parallel processing. *Trends in Neurosciences, 13,* 266–271.

Anderson, J. R. (1995). *Learning and memory*. Singapore: Wiley.

Barto, A. (1995). Adaptive critics and the basal ganglia. In J. Houk, J. Davis, & D. Beiser (Eds.), *Models of information processing in the basal ganglia* (pp. 215–232). Cambridge: MIT Press.

Bissmarck, F., Nakahara, H., Doya, K., & Hikosaka, O. (2005). *Responding to modalities with different latencies*. Paper presented at the Advances in Neural Information Processing Systems, Vancouver, Canada.

Daw, N. D., & Touretzky, D. S. (2002). Long-term reward prediction in TD models of the dopamine system. *Neural Computation, 14,* 2567–2583.

Doya, K. (1999). What are the computations of the cerebellum, the basal ganglia and the cerebral cortex? *Neural Networks, 12,* 961–974.

Doya, K. (2000). Reinforcement learning in continuous time and space. *Neural Computation, 12,* 219–245.

Doyon, J., & Benali, H. (2005). Reorganization and plasticity in the adult brain during learning of motor skills. *Current Opinion in Neurobiology, 15,* 161–167.

Doyon, J., Owen, A. M., Petrides, M., Sziklas, V., & Evans, A. C. (1996). Functional anatomy of visuomotor skill learning in human subjects examined with positron emission tomography. *European Journal of Neuroscience, 8,* 637–648.

Floyer-Lea, A., & Matthews, P. M. (2005). Distinguishable brain activation networks for short- and long-term motor skill learning. *Journal of Neurophysiology, 94,* 512–518.

Foulkes, A. J., & Miall, R. C. (2000). Adaptation to visual feedback delays in a human manual tracking task. *Experimental Brain Research, 131,* 101–110.

Georgopoulos, A. P., Kettner, R. E., & Schwartz, A. B. (1988). Primate motor cortex and free arm movements to visual targets in three-dimensional space: II. Coding of the direction of movement by a neuronal population. *Journal of Neuroscience, 8,* 2928–2937.

Haruno, M., Wolpert, D. M., & Kawato, M. (2001). Mosaic model for sensorimotor learning and control. *Neural Computation, 13,* 2201–2220.

Hikosaka, O., Nakahara, H., Rand, M. K., Sakai, K., Lu, X., Nakamura, K., et al. (1999). Parallel neural networks for learning sequential procedures. *Trends in Neurosciences, 22,* 464–471.

Hikosaka, O., Nakamura, K., Sakai, K., & Nakahara, H. (2002). Central mechanisms of motor skill learning. *Current Opinion in Neurobiology, 12,* 217–222.

Houk, J. C., Adams, J. L., & Barto, A. G. (1995). A model of how the basal ganglia generate and use neural signals that predict reinforcement. In J. C. Houk, J. L. Davis, & D. G. Beiser (Eds.), *Models of information processing in the basal ganglia* (pp. 249–270). Cambridge: MIT Press.

Jacobs, R., Jordan, M., & Barto, A. (1991). Task decomposition through competition in a modular connectionist architecture: The what and where in vision tasks. *Cognitive Science, 15,* 219–250.

Jueptner, M., Frith, C. D., Brooks, D. J., Frackowiak, R. S., & Passingham, R. E. (1997). Anatomy of motor learning: II. Subcortical structures and learning by trial and error. *Journal of Neurophysiology, 77,* 1325–1337.

Kitazawa, S., Kohno, T., & Uka, T. (1995). Effects of delayed visual information on the rate and amount of prism adaptation in the human. *Journal of Neuroscience, 15,* 7644–7652.

Kitazawa, S., & Yin, P. B. (2002). Prism adaptation with delayed visual error signals in the monkey. *Experimental Brain Research, 144,* 258–261.

Kording, K. P., & Wolpert, D. M. (2004). Bayesian integration in sensorimotor learning. *Nature, 427,* 244–247.

Kording, K. P., & Wolpert, D. M. (2006). Bayesian decision theory in sensorimotor control. *Trends in Cognitive Sciences, 10,* 319–326.

Liu, D., & Todorov, E. (2007). Evidence for the flexible sensorimotor strategies predicted by optimal feedback control. *Journal of Neuroscience, 27,* 9354–9368.

McClure, S. M., Ericson, K. M., Laibson, D. I., Loewenstein, G., & Cohen, J. D. (2007). Time discounting for primary rewards. *Journal of Neuroscience, 27,* 5796–5804.

Miall, R. C., & Jackson, J. K. (2006). Adaptation to visual feedback delays in manual tracking: Evidence against

the Smith Predictor model of human visually guided action. *Experimental Brain Research, 172,* 77–84.
Miall, R. C., Weir, D. J., & Stein, J. F. (1985). Visuomotor tracking with delayed visual feedback. *Neuroscience, 16,* 511–520.
Montague, P. R., Dayan, P., & Sejnowski, T. J. (1996). A framework for mesencephalic dopamine systems based on predictive Hebbian learning. *Journal of Neuroscience, 16,* 1936–1947.
Nakahara, H., Doya, K., & Hikosaka, O. (2001). Parallel cortico-basal ganglia mechanisms for acquisition and execution of visuomotor sequences—A computational approach. *Journal of Cognitive Neuroscience, 13,* 626–647.
Ogawa, K., Inui, T., & Sugio, T. (2007). Neural correlates of state estimation in visually guided movements: An event-related fMRI study. *Cortex, 43,* 289–300.
Petersen, E. S., van Mier, H., Fiez, A. J., & Raichle, E. M. (1998). The effect of practice on the functional anatomy of task performance. *Proceedings of the National Academy of Sciences, 95,* 853–860.
Pouget, A., Dayan, P., & Zemel, R. (2000). Information processing with population codes. *Nature Reviews Neuroscience, 1,* 125–132.
Pouget, A., Dayan, P., & Zemel, R. S. (2003). Inference and computation with population codes. *Annual Review of Neuroscience, 26,* 381–410.
Schweighofer, N., Shishida, K., Han, C. E., Okamoto, Y., Tanaka, S. C., Yamawaki, S., et al. (2006). Humans can adopt optimal discounting strategy under real-time constraints. *PLoS Computational Biology, 2,* e152.
Tanaka, S. C., Doya, K., Okada, G., Ueda, K., Okamoto, Y., & Yamawaki, S. (2004). Prediction of immediate and future rewards differentially recruits cortico-basal ganglia loops. *Nature Neuroscience, 7,* 887–893.
Todorov, E. (2004). Optimality principles in sensorimotor control. *Nature Neuroscience, 7,* 907–915.
Todorov, E., & Jordan, M. I. (2002). Optimal feedback control as a theory of motor coordination. *Nature Neuroscience, 5,* 1226–1235.
Weiss, Y., & Fleet, D. J. (2002). Velocity likelihoods in biological and machine vision. In R. Rao, B. Olshausen, & M. S. Lewicki (Eds.), *Statistical theories of the cortex* (pp. 77–96). Cambridge: MIT Press.