

Experimental and educational platforms for studying architecture and tradeoffs in human sensorimotor control

Quanying Liu¹, Yorie Nakahira¹, Ahkeel Mohideen¹, Adam Dai¹,
Sunghoon Choi¹, Angelina Pan¹, Dimitar M. Ho¹ and John C. Doyle¹

Abstract—This paper describes several surprisingly rich but simple demos and a new experimental platform for human sensorimotor control research and also controls education based on an off-the-shelf gaming platform. The platform safely simulates a canonical sensorimotor task of riding a mountain bike down a steep, twisting, bumpy trail using a standard display and inexpensive gaming steering wheel with a force feedback motor. We use the platform to verify our theory, presented in a companion paper, about how component hardware speed-accuracy tradeoffs (SATs) in control loops impose corresponding SATs at the system level, but also how effective architectures mitigate the deleterious impact of hardware SATs through layering and “diversity sweet spots (DSSs). Specifically, we measure the impacts on system control performance of disturbances and delays, quantization, and noise in feedback loops, both within the subjects nervous system and added externally via software in the platform. This provides a remarkably rich test of the theory, which is consistent with all preliminary data. Moreover, as the theory predicted, subjects effectively multiplex specific higher layer planning/tracking of the trail using vision with lower layer rejection of unseen bump disturbances using reflexes. In contrast, humans multitask badly on tasks that do not naturally distribute across layers (e.g. texting and driving). The platform is cheap and easy to build and use, and flexible to program for both research and education, yet highlights crucial gaps in both neuroscience and control theory that our new theory closes.

I. INTRODUCTION

A heavily studied and central topic in neuroscience is speed-accuracy tradeoffs (SATs) [1–5]. At the neuron-component level, the resource limitations (space and metabolic costs) of the brain impose severe speed/accuracy constraints in neural signaling [6], as well as in the muscle actuation [7–9]. At the system level, the tradeoffs associated with SAT hard limits are extensively studied with a variety of experiments [10–13] and mathematical theory [3], [4]. Yet there is little attention to the theory or experiments that can connect system-level SATs with SATs within the underlying nerve components.

A companion paper [14] develops a theoretical framework to connect the system-level and component-level SATs. The results suggest theoretically optimal layering creates “diversity sweet spots (DSSs) to effectively layer sensorimotor control with appropriate diversity in neurons/muscles to achieve systems that are *both* fast and accurate despite being built from components that individually are not.

Motivated by the theory, here we present an experimental platform, some preliminary behavioral results to validate the theory and the relevant neuroscience details for DSSs. In

particular, our platform allows us to study the following three questions and potentially more.

1) *Combining component and system levels*: The system level performance for sensorimotor control is supported by the underlying component hardware, such as lower-layer sensors and muscles and connecting nerves, [7–9] and higher-layer nerves in the spinal cord and in cortex [15], [16]. Each component has its own speed and accuracy in the control loops. Inspired by the theory, developing an experimental platform involving these components in each control layer, and allowing us to manipulate the speed and accuracy in these components, is essential for studying the link between the system-level performance and the component-level features in human sensorimotor control.

2) *DSSs at component-hardware and system levels*: A huge diversity in the size and number of axons has evolved and exists across different sensorimotor nerves. For instance, axon diameters vary by more than 200 times and the cross-sectional area of nerves differ much more. The neural signaling delay is diverse, as both the spiking rate of neurons, and the propagation speed of an axon scales with axon radius [6], [17], whereas bandwidth scales the opposite (to be formalized below). However, it was unclear how the human sensorimotor system achieves the system-level robustness through DSSs, given the constraints on SATs at the component level [17]. We aim to show experimental evidence to connect the DSSs across levels.

3) *Layered architecture in sensorimotor control*: Human sensorimotor control system is extremely robust [18], although the sensing is distributed, sparse, quantized, noisy and delayed [19], [20]; the computing in the central nervous system is slow; [6]; and the muscle actuation fatigues and saturates [21]. Effective layered architectures have evolved, to integrate the higher layers of goals, plans, decisions with lower layers of sensing, reflex, and action [22]. Take riding a mountain bike as an example. Two control layers are involved in this task: the plan layer and the reflex layer. For the visible disturbances (*i.e.* the trail), we make a plan before the disturbance arrives. For the invisible disturbances (*i.e.* small bumps), the control heavily relies on reflexes. The layered architecture manages slow or inaccurate hardware, but facilitates learning, adaptation, augmentation and teamwork. However, how to integrate the sensing / communication / computation / actuation component-hardware constraints with the plan / reflex layers in the human sensorimotor system has been unclear. We show that the effective layering is achieved through DSSs within/across layers and levels in

¹ California Institute of Technology

a way that fits known neural anatomy strikingly well.

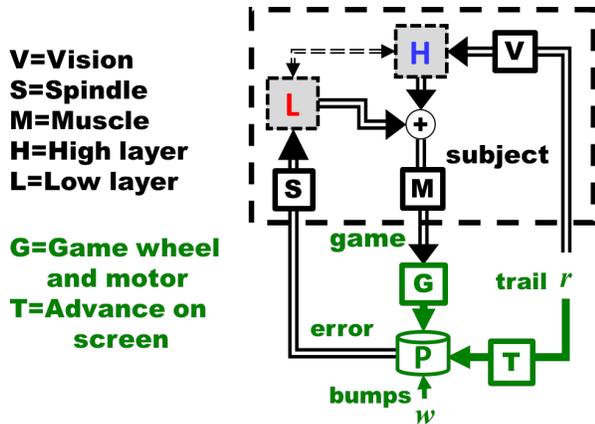


Fig. 1: Basic block diagram for theory and experimental platform with subject and gaming wheel with motor. Each box is a component that communicates or computes and has potentially both delay and quantization, including within the game in G. The advance warning T is also implemented on a computer screen with vision.

To answer these questions, a platform with the control elements (e.g. delay, quantization, disturbance and feedback) is needed. Here we focus on our new, inexpensive and easy-to-use experimental platform that illustrates and tests the new theory that formalizes and explains these connections (Fig. 1). The details for the theory is reviewed in a companion paper [14]. Because it is impossible to noninvasively manipulate the internal delay and data rate in neural signalling, we find an alternative to add the external delay and quantization in actuator which generates an identical effect to simulate the hardware SATs in nerves.

Our sophisticated and versatile platform is a video game that safely simulates riding a mountain bike down a steep, twisted, bumpy trail using a standard display and gaming steering wheel. The virtual trail scrolls down a PC screen which can vary in speed, turns, and visual look ahead (and thus advance warning or delay). Subjects can see the trail and turn the wheel to track it with minimum error, while an internal motor can torque the wheel to mimic invisible bumps in the trail (Fig. 2). The “trail planning and tracking uses vision and advanced warning, while the “bumps disturbances are compensated for with delayed reflex feedback control. Subjects must cope with delays, quantization, disturbances, noise, and component sparsity, all captured in the theory, in both their own nervous system and, to more thoroughly verify the theory noninvasively, when added in the experimental hardware.

In this paper, we highlight the experimental platform while keeping the essence of the theory. In Section II, we first describe the framework of the platform and introduce the component hardware parameters that the platform can manipulate. We will also describe and demonstrate the goal of this game platform. In Section III, we apply the platform for testing the effects of delay in either the visual feedback

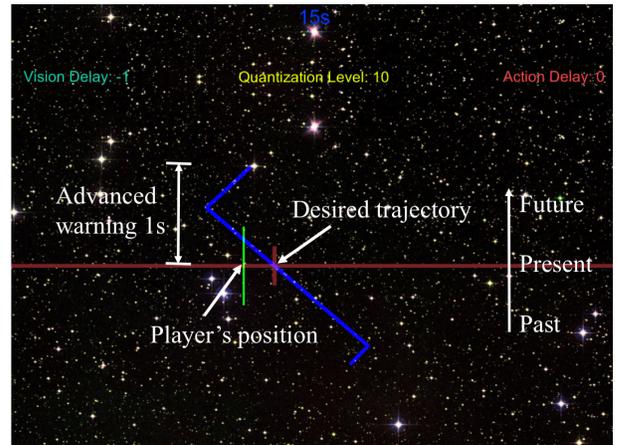


Fig. 2: The video monitor interface for the game. Players see a winding trail scrolling down the screen at a fixed speed, and thus with a fixed look-ahead time T, both of which can be varied widely. The player aims to minimize the error between the desired trajectory and their true screen position using a gaming steering wheel. Bumps are added using a motor torque in the wheel. Experiments can be done with just bumps or trails, or both together, and with varying trail speed and/or T, and with additional quantization and/or delay in the map from wheel position to player screen position.

or the actuator, the speed-accuracy tradeoffs and the effective layered architecture in human sensorimotor control. For each test, we present the theoretical model and derived performance bound, the game setting and experimental results, and discuss how the experimental results support the theory. We end the section with a variety of simple demos involving vision and vestibular-ocular reflex (VOR) that also illustrate principles of both neuroscience and control. In Section IV, we shortly summarize the neuroscience details with a discussion that connects the experiments and the theory, and we list some potential directions along the way.

Notation: We use $x(t_1 : t_2) = \{x(t_1), x(t_1 + 1), \dots, x(t_2)\}$ to denote a truncated sequence. The ℓ_∞ norm of a sequence x is defined as $\|x\|_\infty := \sup_t |x(t)|$. We use $P(x)$ to denote the probability density function of a random variable x , and $P(x|y)$ to denote the conditional probability density function of x given another random variable y . We use $\log(x)$ to denote the logarithm of x to base 2 and $\log_b(x)$ to denote the logarithm of x to base b . We use \mathbb{Z}/\mathbb{Z}_+ and \mathbb{R}/\mathbb{R}_+ to denote the set of all/non-negative integers and the set of all/non-negative real numbers, respectively.

II. PLATFORM AND MODELS

A. Framework of the platform

The video-game platform simulates riding a mountain bike down a steep, twisting, bumpy trail using a standard display and gaming steering wheel. It includes a lower-layer reflex feedback control loop (L in Fig. 1) in charge of the unseen bump disturbance w , and a high-layer advanced plan loop (H in Fig. 1) to see the incoming trail and make a control plan

in advance. Notably, the internal delays and quantizations in neural signalling or muscle contraction encoded in the subject are impossible to control non-invasively. However, we find an alternative way to extensively add the external delay and quantizer with limited data rate in the game. Specifically, the game platform allows us to manipulate the advanced warning or visual delay T_{vis} (T in Fig. 1), along with the action delay T_{act} and action quantizer Q_{act} with data rate R_{act} in the gaming wheel (G in Fig. 1). We haven't yet implemented the quantization Q_{vis} for the visual system. It remains one of our future goals.

While the effects of delay are a combination of internal and external delay, especially when the internal delay is relatively big, such as the delay of flash-evoked responses in the visual cortex (around 120ms in healthy subjects [23]), quantization effects are evaluated only considering the external quantizer. The external quantizer we set in the game has a relatively small data rate (up to 10 bits), whereas the data rates for neural signalling and muscle contraction are much higher, such as the 50ms interspike intervals in motoneuron [24]. Thus, because the data rate for internal neural signalling is so much higher than that of the external quantizer, we neglect the internal quantization to simplify the analysis.

In addition, the trail disturbance $r(t)$ and bump disturbance $w(t)$ can be designed for the game. The design of these disturbances largely depends on the bounds in the feedback control model. We have to separate the worst-case setting and the average-stochastic setting in the game due to the difference in disturbance (see in Section II-B). However, the design of delay and quantization in these two settings are the same.

The output of the game platform is a file including the values of all these manipulated parameters, the control $u(t)$ and the error dynamics $x(t)$. We minimally train subjects to achieve a stable performance. To eliminate the learning effects and uncontrollable variables during task switching, we exclude the first 10 seconds from the data analysis. See details about the hardware, software, input file and output file of the platform in the Appendix.

B. Model and the parameter settings in platform

We consider a simplified feedback control model shown in Fig. 3. The system dynamics is given by

$$x(t+1) = x(t) + w(t) + u(t) + r(t) \quad (1)$$

where $x(t) \in \mathbb{R}$ is the error dynamics, $r(t) \in \mathbb{R}$ is the trail disturbance and $w(t) \in \mathbb{R}$ is the bump disturbance, $u(t) \in \mathbb{R}$ is the control action. $r(t)$ and $w(t)$ are independent.

We characterize the impact of delay and data rate on system performance in two settings:

1) *Worst-case setting*: The uncertainty (noise or changes in the desired trajectory) is bounded. Thus, when we design a game to test the worst-case performance, we set $r(t)$ and $w(t)$ to be periodic to satisfy that the disturbance is infinity-norm bounded, and we analyze the infinity norm of error $\|x\|_\infty$.

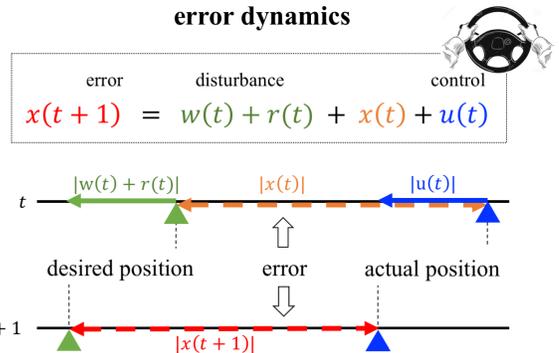


Fig. 3: The error dynamics in the system. Subject can see both the desired position (the trail position) and the actual position (the player's position) at each moment.

2) *Average-case setting*: The uncertainty is drawn from some distributions. For the average-case experiment design, the time at which there is a turn and/or a bump is decided by Gaussian distribution, and we analyze the mean squared error (MSE) with $\frac{1}{T} \left[\sum_{t=1}^T x(t)^2 \right]$.

The worst-case performance is more relevant in risk-adverse situations, while the average-case performance is more relevant in risk-neutral situations. An example of a risk-adverse situation is riding a mountain-bike on a cliff because staying on the cliff is a must for survival even given the worst possible uncertainty. This analysis framework is conceptually simpler when compared to the stochastic case. The worst-case control has been reported in the existing literature for risk sensitive sensorimotor control [25–28]. On the other hand, an example of a risk-neutral situation is riding a mountain bike on a broad field because there exists no risk of falling out of the field. The control goal therefore is to minimize errors from the desired trajectory. To be noted, although the platform supports both the worst-case setting and average-case setting to test different bounds in the theory, we only present the results from the worst-case setting in this paper.

C. Goal of the platform and the experiment design

The experimental platform has two major goals: 1) to test our new theory which connects the component hardware SATs in control loops with system-level SATs. 2) to serve future sensorimotor control research and educational applications.

1) *Validate the new theory*: Through manipulating the external delay and quantization (T_{vis} , T_{act} and Q_{act}), and designing the disturbance, $r(t)$ and $w(t)$, the platform allows us to study the impact of component hardware features on system-level performance. In the worst-case setting, we designed the trail changes in the trajectory $r(t)$ and the bump disturbances $w(t)$ to be ∞ -norm bounded *i.e.* $\|r\|_\infty \leq 1$, $\|w\|_\infty \leq \epsilon$. Particularly, we will resolve the following three questions with specific game designs.

- What are the effects of vision delay and action delay in a single control loop? (*Section III-A*)

We design a game with delay to test the impacts of delay. The sharp turns in the desired trajectory $s(t)$ with the angle $\theta \in \{10, 20, \dots, 80\}$ periodically arrive every 1.5 seconds (the disturbance $r(t) = s(t+1) - s(t)$ with $\|r\|_\infty \leq 0.04$ unit, 1 unit = 100 pixels). No advanced warning is added. Subjects therefore cannot see any future trajectory. To examine the effects of vision delay, we vary the available history of trajectory in the visual feedback. At 0 delay, subjects cannot see the future trajectory, but they can see the full history of past trajectory. However, at 150ms, 300ms, 450ms, 600ms and 750ms of T_{vis} delay, they gradually lose the visual view of the past road. For the action delay test, an external delay is added to the steering wheel, but no vision delay. To be identical to the vision delay test, no advanced warning is provided. The 0 action delay is the baseline setting; then 150ms, 300ms, 450ms, 600ms or 750ms of action delay, T_{act} , is added, all while keeping 0 vision delay so that the results are comparable between these two tests. Each setting in the game lasts for 30 seconds, so the total experiment is 6 minutes.

- How do component hardware SATs in layers impose the system-level SATs? (*Section III-B*)

To test SATs in the higher-level plan layer, we design a game with visual advanced warning/delay and quantization. To test performance in the SATs with limited data rate (quantization) and advanced warning/time delay, we added either quantization in the actuator, or additional advanced warning/time delay in the visual input, or both. The quantization Q_{act} is put on the output from the steering wheel (the angle of the wheel), where the data rate, R , is set to 1, 2, ..., 7 bits, respectively. The corresponding advanced warning/time delay, T_{vis} , is -800, -600, ..., 400ms, respectively. The negative value means advanced warning in the visual input, whereas the positive value means delay. For the T-R tradeoff settings, $T = 200(R - 5)$. Each setting lasted 30 seconds, so it takes $7 \times 30 \times 3 = 630$ seconds in total. To test SATs in the lower-level reflex layer, we design a game with bumps, action delay and quantization. The bump disturbance $w(t)$ arrives every 2 seconds, with a maximum 100 units of torque. Similar to the setting for SATs in the advanced plan layer, we externally add either a quantizer with limited data rate R , or a delay T , or both in the steering wheel. The R is set to 1, 2, 3, 4 bits, respectively, and T is set to 0, 200, 400, 600ms, respectively. For the T-R tradeoff test, $T = 200(R - 1)$. Each setting lasted 30 seconds (in total $4 \times 30 \times 3 = 360$ seconds).

- How does the layering architecture effectively help the sensorimotor control? (*Section III-C*)

To be more realistic and natural, we developed a bump and trail dual-task game in the platform. We either add bump disturbance $w(t)$, or trail disturbance $r(t)$, or bump and trail dual disturbance $w(t) + r(t)$. $w(t)$ and $r(t)$ are independent. The bump effects were generated by a 0.5s constant torque to the steering wheel. To be noted, we did not intentionally add external delays in the game, and we used the maximum

quantization ($R=10$ bits).

2) *Beneficial to research and education*: The platform is flexible to program, and easy to extend to other tasks and games. Both the source code and the executable file are open-access in our website. It has a user-friendly graphic user interface (GUI) for naive users. It also supports the higher-level individualized task design. We hope the game platform will have versatile applications for both research and education.

III. EXPERIMENTS AND INTERPRETATIONS

In this section, we first study the impact of delay in a single control loop. It is also uses the simplest example with one variable in the game to show details of the game design with the platform (*Section III-A*). Then we further investigate the component level SATs in the advanced plan layer with a trail task and in the reflex layer with a bump task (*Section III-B*), and the benefits of the effective layering architecture with a bump and trail dual task (*Section III-C*).

A. Delay in sensorimotor control

Delay is an essential topic for both control engineering and neuroscience [23], [29]. Delay in human sensorimotor control is inevitable, existing in each step: sensing, communication, computation and actuation. In this section, using the experimental platform combined with feedback control theory, we investigate how visual delay and action delay influence human sensorimotor control performance.

1) *Simple model for delay in control*: In this section, we study the impact of delay in sensing (vision) or actuation (muscle). We model the control action by

$$u(t + T) = \mathcal{K}(x(0 : t), r(0 : t), u(0 : t + T - 1)). \quad (2)$$

The game starts with zero initial condition, *i.e.*, $x(0) = 0$. The controller \mathcal{K} generates the control command $u(t)$ using the full information on the histories of state, disturbance, and control input. The control command is executed with delay $T \geq 0$.

Sensorimotor control in the risk-aware setting [28] motivates the use of L_1 optimal control, and as such, our goal is to verify the following robust control problem:

$$\inf_{\mathcal{K}} \sup_{\|r\|_\infty \leq 1} \|x\|_\infty \quad (3)$$

This problem admits a simple and intuitive solution. In particular, the optimal cost is given by

$$\inf_{\mathcal{K}} \sup_{\|r\|_\infty \leq 1} \|x\|_\infty = T. \quad (4)$$

This optimal cost is achieved by the worst-case control policy $u(t + T) = -r(t)$, which yields

$$\inf_{\mathcal{K}} \sup_{\|r\|_\infty \leq 1} \|u\|_\infty = 1. \quad (5)$$

Interestingly, the control effort is not a function of the delay.

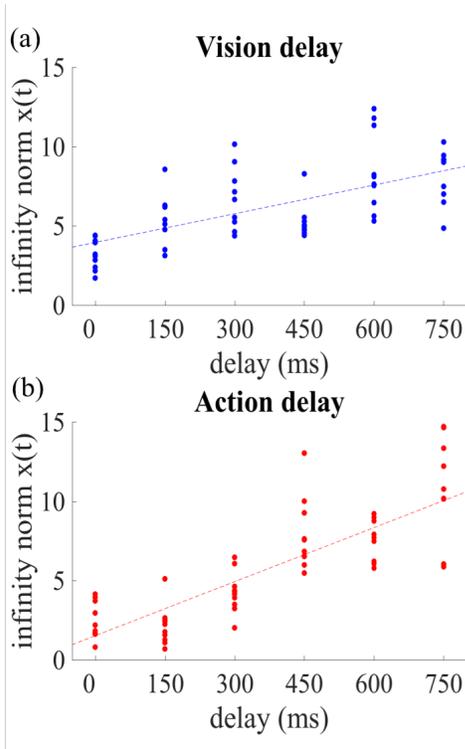


Fig. 4: The experimental results for effects of delay in visual feedback and in the action, respectively. We present the performance $\|x\|_\infty$ for each 2s.

2) Trail with delay game and experimental findings:

The experimental results for the trail with delay game (see Section II-C for the game design) are illustrated in Fig. 4.

The one-way analysis of variance (ANOVA) for $\|x\|_\infty$ shown in Fig. 4(b) found significant effects of T for both vision delay ($F = 11.49$, $p < 1e-7$) and action delay ($F = 28.66$, $p < 1e-14$). It means that the error significantly increases with an increase of delay. The generalized linear model was applied to fit the data. It showed that $\|x\|_\infty = 0.006T + 3.956$ for the vision delay, and $\|x\|_\infty = 0.011T + 1.528$ for the action delay. The results indicate that the error has a linear relation with the delay in a control loop. It is well in line with the prediction from the theoretical model in Eq.(4). Although the theory doesn't distinguish the impacts of vision delay from those of action delay on performance, the experimental data does show a smaller slope with increases in vision delay. This might be because the design of the trail is not random, and the subject can partially predict the future trajectory.

B. Speed-accuracy tradeoffs in a single layer

The impact of the SATs in nerve signaling on the SATs in sensorimotor control performance has been characterized in our previous paper [17] and is reviewed in our companion paper [14]. In this section, we validate the theoretical results using a mountain-bike game in the experimental platform.

1) *Model connecting the component-level and system-level SATs:* Here, we demonstrate the impact of delay and

quantization in the sensorimotor control loop. In the game, the control action is generated by the following feedback loop with communication constraints; we therefore model:

$$u(t+T) = Q(K_t(x(0:t), u(0:t-1))). \quad (6)$$

where $K_t : (\mathbb{R}^{t+1}, \mathbb{R}^t) \rightarrow \mathbb{R}$ is a controller, and $Q : \mathbb{R} \rightarrow S$ is a quantizer with data rate $R \geq 1$, i.e. S is a finite set of cardinality 2^R . Here, the net delay is composed of internal delays in the human sensorimotor feedback and the delays externally added. The disturbance is ∞ -norm bound and, without loss of generality, $\|r\|_\infty \leq 1$.

The worst-case state deviation is lower-bounded by [17]:

$$\sup_{\|r\|_\infty \leq 1} \|x\|_\infty \geq T + \frac{1}{2^R - 1} \quad (7)$$

and the minimum control effort is given by

$$\sup_{\|r\|_\infty \leq 1} \|u\|_\infty \geq \left(1 + \frac{1}{2^R - 1}\right) \left(1 - \frac{1}{2^R}\right) \quad (8)$$

Here equality in (7) and (8) can be obtained at the same time.

2) Experiments showing SATs in a simple control loop:

We present the experimental results both for the SATs test in the plan feedback layer (Fig. 5), and for the SATs test in the reflex feedback layer (Fig. 6). Both layers show clear system-level tradeoffs resulting from the hardware tradeoffs. Although the infinity norm error heavily relies on the worst performance (one mistake would largely elevate the error measurement), our results surprisingly show that the effects of the T only test and the Q only test is roughly equivalent to the sum of the infinity error from the T and Q test after systematically subtracting the intrinsic error. It validates the Eq. (7) in theory that the control error in T and R is the sum of T term and R term.

Observed from the data, the tradeoff sweet spot for the plan layer is $R = 4$ bits and $T = -200$ ms when the component tradeoff is $T = 200(R - 5)$, whereas the tradeoff sweet spot for the reflex layer is $R = 3$ bits and $T = 400$ ms when the component tradeoff is $T = 200(R - 1)$. To be noted, the component level tradeoffs are not the same in the plan layer and the reflex layer in the experiments, because we try to test both the advance warning and delay in the plan layer. One future objective is to test the plan layer with the same component level tradeoff as the reflex layer.

Fitting the data with the quantization term in Eq.(7), $\|x\|_\infty = \alpha/(2^R - 1)$, we obtain $\alpha = 8.80$ for the higher-layer plan loop, and $\alpha = 7.83$ for the lower-layer reflex loop in the quantization only test. It indicates that the plan layer is more sensitive to the limited data rate, compared to the reflex layer. Fitting the data with the delay term in Eq.(7), $\|x\|_\infty = \beta T$, we obtain $\beta = 0.12$ for the reflex layer, and $\beta = 0.04$ for the plan layer, suggesting that the performance of the reflex layer is more sensitive to the delay. A increased delay T has deleterious impacts on the bump rejection. Together, the results demonstrate “diversity sweet spots at the component hardware level cross layers in human sensorimotor control system.

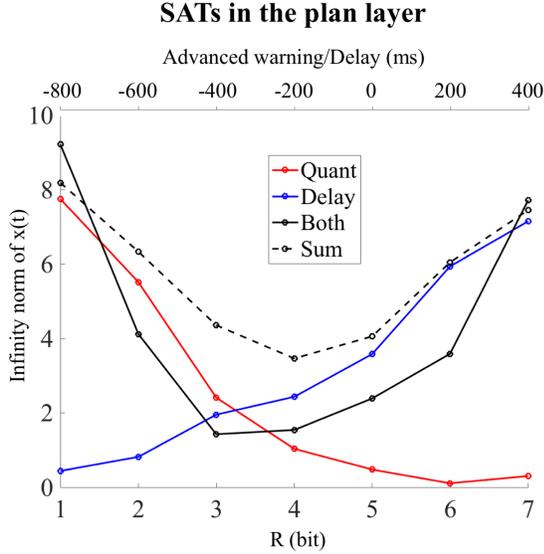


Fig. 5: The experimental results for SATs in the plan layer. For the visualization, we subtract the intrinsic error systematically. For the results from the delay only and Quantizationdelay tests, the corresponding delay $T = 200(R - 5)$.

C. Effective layered architecture in sensorimotor control

Human sensorimotor control is an extremely robust system. It involves multiple control loops in layered architectures, rather than a simple one, as well as a huge diversity of component hardware to manage slow or inaccurate hardware [22]. The effectiveness of layered architectures can be observed in many sensorimotor control tasks. To test the robust control in multiple control loops and the effective layered architecture in human sensorimotor control, we have implemented the 'bump and trail' scenario in our platform, which mimics riding a mountain bike down a steep, twisting, bumpy trail .

1) *Theoretical model for the layered architecture:* In the game, the system dynamics in Eq.(1) have the ∞ -norm bounded disturbance, with $\|r\|_\infty \leq 1$, $\|w\|_\infty \leq \epsilon$. The actuation action is generated through two controllers, a lower-layer reflex loop mainly in charge of $w(t)$ (e.g. human reflection responding to the bumps), and a high-layer advanced plan loop mainly in charge of $r(t)$ (e.g. our planning for the future road). The theory considers two types of layered architectures: with/without sharing the disturbance information between the two controllers. The layered architecture with shared information is defined by

$$\begin{aligned} u_L(t + T_L) &= L(x(0 : t), u(0 : t + T_L - 1), w(0 : t)) \\ u_H(t + T_H) &= H(x(0 : t), u(0 : t + T_H - 1), r(0 : t + T_a)) \\ u(t) &= u_L(t) + u_H(t). \end{aligned} \quad (9)$$

Here, L, H are the controllers, and Q_L, Q_H are the quantizers with data rates R_L, R_H , respectively.

The layered architecture without shared information is

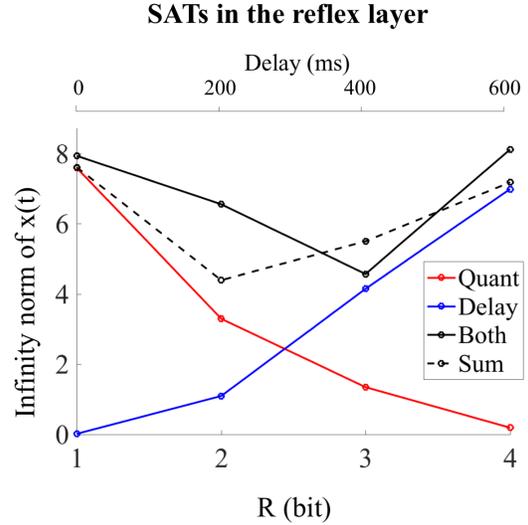


Fig. 6: The experimental results for SATs in the reflex layer. For the visualization, we only show R in the x axis. For the results from the delay only and Quantizationdelay tests, the corresponding delay $T = 200(R - 1)$ as x axis.

defined by

$$\begin{aligned} u_L(t + T_L) &= L(u(0 : t + T_L - 1), w(0 : t)) \\ u_H(t + T_H) &= H(u(0 : t + T_H - 1), r(0 : t + T_a)) \\ u(t) &= u_L(t) + u_H(t). \end{aligned} \quad (10)$$

We pose the robust control problem as follows:

$$\inf_{\|r\|_\infty \leq \epsilon, \|w\|_\infty \leq 1} \sup \|x\|_\infty, \quad (11)$$

where the infimum is taken over the control policies of the form (9) or (10). The worst-case state-deviation of the system with shared information is given by

$$\left\{ T_\ell + \frac{1 - 2^{-R_\ell(T_h - T_\ell)}}{2^{R_\ell} - 1} + \frac{1}{2^{R_\ell + R_h} - 1} \right\} (1 + \epsilon). \quad (12)$$

The worst-case state-deviation of the system without shared information is given by

$$\left\{ T_\ell + \frac{1}{2^{R_\ell} - 1} \right\} \epsilon + T_h + \frac{1}{2^{R_h} - 1}. \quad (13)$$

The experiments allow to test whether there is shared disturbance information between two layers or not.

2) *Bump and trail dual task to test the plan/reflex layering:* Our experimental results (Fig. 7) demonstrate that the error from the bump and trail session positively correlated with the sum of the error from the bump only session and the trail only session (Pearson correlation, $R = 0.57$, $p < 0.0001$), and they showed no significant difference (paired ttest, $t = 0.21$, $p = 0.83$). The results confirm that the sensorimotor control can be decomposed to a reflex layer and an advanced plan layer. It fits the theory with no shared information case in Eq.(13), rather than Eq.(12). It indicates that the plan layer and the reflex layer can

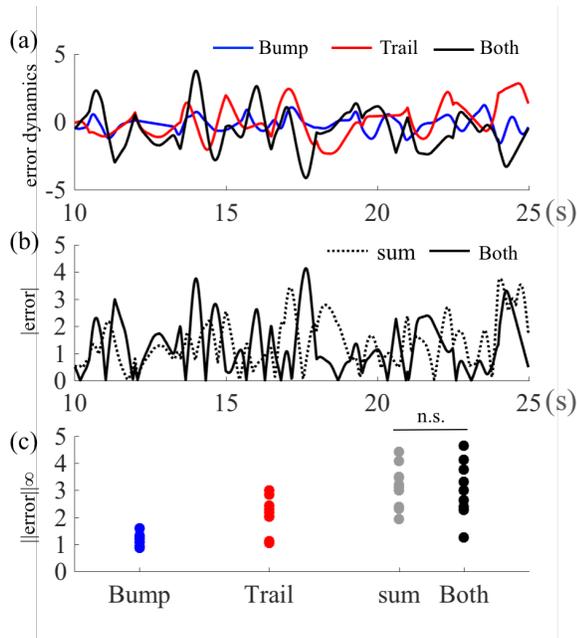


Fig. 7: The experimental results bump and trail task: (a) the error dynamics from bump only task, trail only task, both task; (b) absolute error; (c) infinity norm error. One dot denotes the infinity norm error in 2 seconds.

be decomposed and analyzed separately. In this case, the multitasking performance is equal to, not bigger than, the sum of each single task (i.e. tracking the trail and rejecting the bump). On the contrary, people usually multitask badly due to dual-task inference [30]. This excellent bump and trail dual task performance can be explained by the effective layering architecture in human sensorimotor control. The two feedback loops jointly control the error $x(t)$ according to Eq.(10), where the advanced plan layer handle the disturbance $r(t)$ from the curvature of the desired trajectory and the delayed reflex layer due to road bumps $w(t)$.

The dual-task test results, combined with the SATs tests in Section III-B, suggest that the effective layered architecture helps us multiplex via the DSSs across layers. The reflex feedback typically operates in the regime of delayed reaction because bumps are often felt by reflexes after the bike hits them. The planning feedback typically operates in the regime of advanced planning because the future trajectory is often seen in advance. By analogous discussion, we would consider the reflex feedback has better performance when it is built to have small signaling delay at the expense of a low signaling rate. On the other hand, the planning feedback when built to have a large signaling rate at the expense of large signaling delay. Indeed, the visual cortex feedback is much slower (greater than 200ms of delay) than the VOR feedback (approximately 10ms of delay) [31]. The actual time delay in visual cortex can be assessed by the following task. Subjects were asked to steer the wheel as quickly as possible when they saw a target. The vision delay was measured by the difference between the time to present the

visual target and the initial time to start steering (N=1, 198 trials, mean=295.1ms, ste=3.8).

3) *Other tests to illustrate DSSs*: There is another simple way to qualitatively show the DSSs across layers. Considering object tracking, watch your hand placed in front of your face: moving the hand left and right with increasing frequency while holding the head still (Test 1); or shaking the head (in a 'no' pattern) at growing frequencies while holding the hand still (Test 2). In Test 1, the hand starts to blur at around 1-2 Hertz due to delays in tracking. In Test 2, blurring due to the inability to compensate for fast head motion occurs at a much higher frequency. This difference illustrates that the visual cortex feedback responsible for Test 1 (object tracking) has lower levels of tolerable delays than the VOR feedback responsible for Test 2 (head motion compensation). Though slower, the visual cortex feedback is known to be accurate, whereas the VOR feedback has limited accuracy. The inaccuracy of VOR can be observed from the fact that standing on one leg with your eyes closed is more difficult than with the eyes open. This is because, when the eyes are closed, balancing cannot access the accurate visual information and relies heavily on the VOR feedback. These simple tests suggest that the system layers with appropriate diversity of SATs in control loops (number of neurons and cross-sectional area of nerves) to achieve both fast and accurate.

IV. DISCUSSION AND FUTURE DIRECTIONS

In this paper, we present an experimental platform for studying the layering architecture and tradeoffs in human sensorimotor control. Our experiments focused on the worst-case scenario because of its theoretical simplicity, although it is experimentally harder for we cannot really simulate to ride a mountain bike on a cliff.

A. Neuroscience perspectives of experiment and platform

We demonstrate that the system-level tradeoffs in human sensorimotor control result from SATs at component-hardware level. We show clear component-level DSSs to favor the system-level SATs cross control layers (Fig. 5 and 6). Furthermore, the results showed that the reflex layer is sensitive to the delay, whereas the advanced plan requires higher accuracy, which are consistent with our theory. Our experiments illustrate the theory that connects the system and component level tradeoffs.

Moreover, the experimental data for the bump and trail dual task shows that optimal controllers with separate layers for trails and bumps can multiplex these tasks well, with errors in the bump with trail task roughly equivalent to, not bigger than, the sum of the errors in the trail only and bump only task, and healthy human subjects achieve this with minimal training (Fig. 7). In contrast, humans multitask badly on the tasks that don't naturally distribute across layers, regardless of whether the tasks are simple (e.g., perceptual discriminations) [30] or complex (e.g., driving and talking on the cell phone) [32]. Our experiments support the theory that the effectively layered sensorimotor control with

appropriate diversity in components to achieve systems both fast and accurate despite being built from components that individually are not.

More importantly, the game platform is inexpensive, easy to use. It allows diverse component hardware features to be flexibly extended in the software. It noninvasively provides a more thorough test of the control theory with experiments. This user-friendly platform has potential both for further research on human sensorimotor control and for control education.

B. Future directions

Our theory and platform are only a beginning of a new regime. We serve much broader interests of SATs and layered architectures in brain, from control engineers to neuroscientists and system biologists. Along this line, we would like to propose some potential research questions that could be studied with our experimental platform in the future:

- Whether is the optimal control policy obtained the control theory applied in real practice? In this paper, we only examined the system performance. It is valuable to investigate the control policy in human sensorimotor control.
- How do the tradeoffs in speed / accuracy / saturation / energy cost in muscles benefit from the DSSs?
- How does the SATs in the high-layer plan/decision making support human/animals' decision strategies across complex environment under uncertainty, limited information, and risks?
- How does the human sensorimotor system tolerate the noise in control loops?
- What are the effects of learning/adaptation in different control layers? Do the tradeoffs exist between fast learning and fast forgetting, between efficiency and plasticity?

In summary, we provide a cheap, easy to use and flexible to program platform not only for validating our theory, but also serving for both future research and education. We believe it will help to address crucial gaps in both neuroscience and control education.

REFERENCES

- [1] V. Kostina and B. Hassibi, "Rate-cost tradeoffs in control," in *Communication, Control, and Computing (Allerton), 2016 54th Annual Allerton Conference on*. IEEE, 2016, pp. 1157–1164.
- [2] A. J. Nagengast, D. A. Braun, and D. M. Wolpert, "Risk sensitivity in a motor task with speed-accuracy trade-off," *Journal of neurophysiology*, vol. 105, no. 6, pp. 2668–2674, 2011.
- [3] R. P. Heitz, "The speed-accuracy tradeoff: history, physiology, methodology, and behavior," *Frontiers in neuroscience*, vol. 8, p. 150, 2014.
- [4] R. Bogacz, E.-J. Wagenmakers, B. U. Forstmann, and S. Nieuwenhuis, "The neural basis of the speed-accuracy tradeoff," *Trends in neurosciences*, vol. 33, no. 1, pp. 10–16, 2010.
- [5] R. Plamondon and A. M. Alimi, "Speed/accuracy trade-offs in target-directed movements," *Behavioral and brain sciences*, vol. 20, no. 2, pp. 279–303, 1997.
- [6] P. Sterling and S. Laughlin, *Principles of neural design*. MIT Press, 2015.
- [7] C. J. De Luca and Z. Erim, "Common drive of motor units in regulation of muscle force," *Trends in neurosciences*, vol. 17, no. 7, pp. 299–305, 1994.
- [8] V. Brezina, I. V. Orekhova, and K. R. Weiss, "The neuromuscular transform: the dynamic, nonlinear link between motor neuron firing patterns and muscle contraction in rhythmic behaviors," *Journal of neurophysiology*, vol. 83, no. 1, pp. 207–231, 2000.
- [9] S. Li, C. Zhuang, M. Hao, X. He, J. C. Marquez Ruiz, C. M. Niu, and N. Lan, "Coordinated alpha and gamma control of muscles and spindles in movement and posture," *Frontiers in computational neuroscience*, vol. 9, p. 122, 2015.
- [10] P. M. Fitts and J. R. Peterson, "Information capacity of discrete motor responses," *Journal of experimental psychology*, vol. 67, no. 2, p. 103, 1964.
- [11] R. W. Soukoreff and I. S. MacKenzie, "Towards a standard for pointing device evaluation, perspectives on 27 years of fitts law research in hci," *International journal of human-computer studies*, vol. 61, no. 6, pp. 751–789, 2004.
- [12] J. Freeston and K. Rooney, "Throwing speed and accuracy in baseball and cricket players," *Perceptual and motor skills*, vol. 118, no. 3, pp. 637–650, 2014.
- [13] R. van den Tillaar and A. Ulvik, "Influence of instruction on velocity and accuracy in soccer kicking of experienced soccer players," *Journal of motor behavior*, vol. 46, no. 5, pp. 287–291, 2014.
- [14] Y. Nakahira, Q. Liu, N. Bernat, T. Sejnowski, and J. C. Doyle, "Theoretical foundations for layered architectures and speed-accuracy tradeoffs in sensorimotor control," submitted to ACC2019.
- [15] D. B. Lockhart and L. H. Ting, "Optimal sensorimotor transformations for balance," *Nature neuroscience*, vol. 10, no. 10, p. 1329, 2007.
- [16] G. E. Loeb, I. E. Brown, and E. J. Cheng, "A hierarchical foundation for models of sensorimotor control," *Experimental brain research*, vol. 126, no. 1, pp. 1–18, 1999.
- [17] Y. Nakahira, N. Matni, and J. C. Doyle, "Hard limits on robust control over delayed and quantized communication channels with applications to sensorimotor control," in *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*. IEEE, 2015, pp. 7522–7529.
- [18] D. W. Franklin and D. M. Wolpert, "Computational mechanisms of sensorimotor control," *Neuron*, vol. 72, no. 3, pp. 425–442, 2011.
- [19] P. M. Bays and D. M. Wolpert, "Computational principles of sensorimotor control that minimize uncertainty and variability," *The Journal of physiology*, vol. 578, no. 2, pp. 387–396, 2007.
- [20] M. Desmurget and S. Grafton, "Forward modeling allows feedback control for fast reaching movements," *Trends in cognitive sciences*, vol. 4, no. 11, pp. 423–431, 2000.
- [21] J. Blinks, R. Rüdell, and S. Taylor, "Calcium transients in isolated amphibian skeletal muscle fibres: detection with aequorin," *The Journal of Physiology*, vol. 277, no. 1, pp. 291–323, 1978.
- [22] J. C. Doyle and M. Csete, "Architecture, constraints, and behavior," *Proceedings of the National Academy of Sciences*, vol. 108, no. Supplement 3, pp. 15 624–15 630, 2011.
- [23] A. Halliday, W. McDonald, and J. Mushin, "Delayed visual evoked response in optic neuritis," *The Lancet*, vol. 299, no. 7758, pp. 982–985, 1972.
- [24] W. H. Calvin and C. F. Stevens, "Synaptic noise and other sources of randomness in motoneuron interspike intervals," *Journal of neurophysiology*, vol. 31, no. 4, pp. 574–587, 1968.
- [25] P. Whittle, "Risk-sensitive optimal control," 1990.
- [26] H. Nagai, "Bellman equations of risk-sensitive control," *SIAM journal on control and optimization*, vol. 34, no. 1, pp. 74–101, 1996.
- [27] T. D. Sanger, "Neuro-mechanical control using differential stochastic operators," in *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*. IEEE, 2010, pp. 4494–4497.
- [28] —, "Risk-aware control," *Neural computation*, 2014.
- [29] W. M. Smith and K. F. Bowen, "The effects of delayed and displaced visual feedback on motor control," *Journal of motor behavior*, vol. 12, no. 2, pp. 91–101, 1980.
- [30] H. Pashler, "Dual-task interference in simple tasks: data and theory," *Psychological bulletin*, vol. 116, no. 2, p. 220, 1994.
- [31] C. Bodelon, M. Fallah, and J. H. Reynolds, "Temporal resolution of the human visual system for processing color, orientation, and color/orientation conjunctions," *Journal of Vision*, vol. 5, no. 8, pp. 758–758, 2005.
- [32] P. E. Dux, M. N. Tombu, S. Harrison, B. P. Rogers, F. Tong, and R. Marois, "Training improves multitasking performance by increasing the speed of information processing in human prefrontal cortex," *Neuron*, vol. 63, no. 1, pp. 127–138, 2009.

APPENDIX

Here we describe the details about how to install the platform, how to design a new game with our experimental platform, and some important notes for using it to broadly study human sensorimotor control.

A. *Hardware and Software of the platform*

The hardware for the platform requires a Logitech steering wheel as a controller, a monitor to present the visual input and visual feedback, and a computer to decode the controller command, and synchronize with game. To run the game, Unity ¹, Visual Studio ², and the Logitech steering wheel SDK ³ have to be installed. The software and source code for platform, and MATLAB code to design the input of the game and analyze the output of the game can be downloaded in our website.

B. *How to design a game in the platform*

The game requires an input text file that contains values for all the experimental variables at every time. Each time, and the values of the variables at that time, should be placed on individual rows in the text file. For the Mountain Bike Task, each row of the text file should have a format of "time, path position, quantization level, vision delay, bump size, action delay" with no spaces and no quotations. The position of the path can be any number between -10 and 10 because these represent the maximum boundaries of the playable screen. The quantization level should be a whole number greater than 0. Vision delay is measured in seconds with a positive delay shrinking the amount of visible path and a negative delay providing visible path. The bump size corresponds to the amount of force feedback on the wheel at that time and should be an integer between -100 (max counter-clockwise force) and 100 (max clockwise force) with 0 meaning no force at all. Action delay is also measured in seconds, but should always be either a positive number or 0. The time between the time values of two adjacent rows (time resolution) should at max be 0.01s. The game will play just fine with input files at a higher resolution than 0.01s.

The game will load the information from the input text file before the game begins. It will assume the time on the last line of the input text file to be the total length of the game. When the game finishes, it will create an output text file from information gathered on each update (approx 0.01s). That file contains the following: time, error, trail position, bump size. On each update of the game, the software calculates the error at that time to be the displacement between the player's position and the trail's position (desired position). Trail position in the output file is the trail's position at the current time as read from the input file. The bump size is the size of the bump at that time as read from the input file. Therefore, at every time in the output file, the trail position and bump size should be exactly equal to their

respective values in the input file. On each update, the software calculates the necessary information and creates the formatted string above. At the end of the game, the strings are compiled together, each to its own row, and then saved in an output text file.

¹<https://unity3d.com/>

²<https://visualstudio.microsoft.com/>

³<https://www.logitechg.com/en-us/innovation/developer-lab.html>