

# Neuromechanics of a Button Press

Antti Oulasvirta, Sunjun Kim

Department of Communications and Networking  
Aalto University, Finland  
antti.oulasvirta@aalto.fi, sunjun.kim@aalto.fi

Byungjoo Lee

Graduate School of Culture Technology  
KAIST, Republic of Korea  
byungjoo.lee@kaist.ac.kr

## ABSTRACT

To press a button, a finger must push down and pull up with the right force and timing. How the motor system succeeds in button-pressing, in spite of neural noise and lacking direct access to the mechanism of the button, is poorly understood. This paper investigates a unifying account based on neuromechanics. Mechanics is used to model muscles controlling the finger that contacts the button. Neurocognitive principles are used to model how the motor system learns appropriate muscle activations over repeated strokes though relying on degraded sensory feedback. Neuromechanical simulations yield a rich set of predictions for kinematics, dynamics, and user performance and may aid in understanding and improving input devices. We present a computational implementation and evaluate predictions for common button types.

## ACM Classification Keywords

H.5.2. Information Interfaces and Presentation (e.g. HCI): User Interfaces; Input devices and strategies.

## Author Keywords

Buttons; input devices; neuromechanics; perceptual control; control theory; probabilistic motor control; input engineering.

## INTRODUCTION

This paper contributes to study of buttons as input devices [1, 14, 23, 25, 33, 36, 41, 42, 46, 51, 53, 58, 71]. A button is a transducer that registers motion of a finger, changes the state of a machine, and returns to resting state. We are interested in the simple press: a press that activates the button and pulls up. In contrast, input using buttons has been studied almost exclusively as an aiming task. However, in many, if not most, presses, clicks, and taps, virtually no aiming is involved. As the finger is resting on or hovering on top of the button, the challenge is to produce the appropriate amount of force to contact the button at the right time. Control of timing is important, because we often need to coordinate inputs with computer-controlled events. The use of force, on the other hand, affects ergonomics and success in activating the button. To improve buttons, we need to understand the motor control processes and the effects of design parameters like button dimensions, materials, force curves, activation point, and feedback.

Our long-term objective is a theoretically plausible yet also actionable theory of button-pressing. To this end, we have been investigating an approach based on *neuromechanics*, the study of motor control in biomechanical and neural systems [10, 16, 52, 68]. Combining concepts from biomechanics, control theory, neurosciences, and machine learning, it has been applied to study character animation, robotics, rehabilitation, gait, and grasping.

In this paper, we take first steps toward neuromechanical modeling that can yield realistic predictions for both user performance (success and precision in activating buttons) and moment-by-moment events (e.g., force and velocity profiles) in button-pressing.

## Overview of Approach and Paper

Neuromechanical modeling starts with asking what is the control problem the nervous system must solve. In the case of button-pressing, the motor system must learn to control a “black box” over degraded signals (see Figure 1). This black box consists of not only the button but the finger, muscles, and sensory systems interacting with it physically. Noise and delay in neural pathways and muscles compromise the transmission and execution of this signal. Physical contact with a button occurs in only 100 milliseconds [37], which is too brief for real-time corrections [22]. Hence, button-pressing is an open-loop problem. A complex cascade of physical events follows the motor command. The brief dynamic event, during which the fingertip presses the button and releases it, gives rise to sensory stimulations, such as mechanoreception, proprioception, vision, and audition, each with distinct signal properties. Thus, the motor system is at the mercy of its limited sensory signals as evidence of “what happened.” Using these signals, it should somehow program a more appropriate muscle activation pattern based on limited sensations.

While the elementary aspects of the “black box” can be simulated with mechanics, the neurocognitive principles have not been explored before. In this paper, we discuss how to model muscles, the fingertip (including its tissue, or pulp), the dynamics of the finger and button contacting, and the sensory feedback signals this produces. Our account of the motor controller rests on the convergence of two principles:

**Perceptual control:** With no direct access to the button, the motor system relies on its limited sensations to adapt the motor commands. We formulate a perceptual control objective [8, 11, 19, 55, 56, 57] for button-pressing: the goal of the motor system is to control its own sensations arising as a consequence of the button press. In particular, we model *timing estimates* that a user makes based on noisy sensory signals.

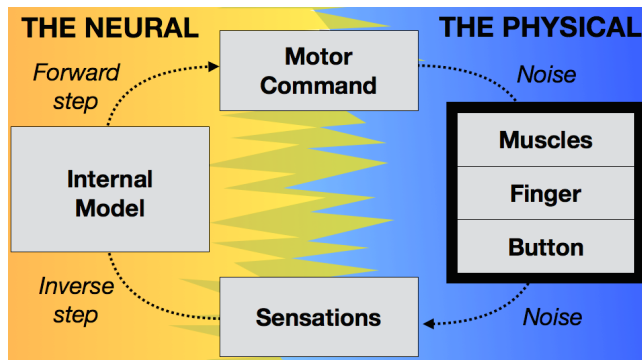
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author.

CHI 2018, April 21–26, 2018, Montreal, QC, Canada

© 2018 Copyright is held by the owner/author(s).

ACM ISBN 978-1-4503-5620-6/18/04.

<https://doi.org/10.1145/3173574.3174082>



**Figure 1.** The neuromechanical problem of button-pressing: To press a button, the motor system must learn how its outputs – mediated by noisy neural pathways – can produce desired effects on a “black box” (right) that it cannot, however, access directly. To this end, it must solve two problems: (1) how to infer noisy sensory signals resulting from a button press and (2) how to choose a motor command that activates the button.

**Probabilistic internal model:** To control a button, the brain must learn to predict the consequences of the actions it takes [13, 26, 35, 39, 40, 62, 64]. We build on previous work on predictive processing and the Bayesian brain hypothesis to posit a probabilistic internal model. It tries to learn to predict the perceptual effects that its open-loop commands have over episodes of attempts with a button. Our implementation uses Gaussian process regression to learn the nonlinear function that maps its commands to perceived responses of the button. A probabilistic model is natural in the face of noisy and incomplete signals. In a forward step, it chooses the motor command that it predicts activates the button at the desired moment in time. In the inverse step, after the button press, it computes an estimate of when the button was activated over the noisy sensory feedbacks it has access to, which is compared against its prediction. As our model of sensory integration, we deploy the cue integration theory [17, 18, 40, 43].

**NEUROMECHANIC** (written in SMALL CAPS to distinguish from neuromechanics, the field) is a computational implementation. It predicts optimal performance under constraints. In particular, it predicts an upper limit to button-pressing performance as bounded by neural, physical, and physiological factors in the model. The model can be used as a workbench for exploring theoretical ideas or comparing button designs. Some aspects of user tasks and preferences can be modeled too, by changing the weights of NEUROMECHANIC’s objective function. For example, we can simulate a light, ergonomic press that requires less force, or presses that are temporally precise.

In the rest of the paper, we review these theoretical ideas and then introduce the computational model. We simulated presses with four common button types (linear, tactile, touch, mid-air). We report simulation results for (1) displacement–velocity patterns, (2) temporal precision and success rate in button activation, and (3) use of force, comparing with effects reported in empirical studies [7, 36, 42, 46, 48, 53, 54, 59, 66, 69]. Over the simulations, we find evidence for the plausibility of the optimality assumption. Finally, we discuss the theoretical position of neuromechanics, outline limitations, and propose topics for future work.

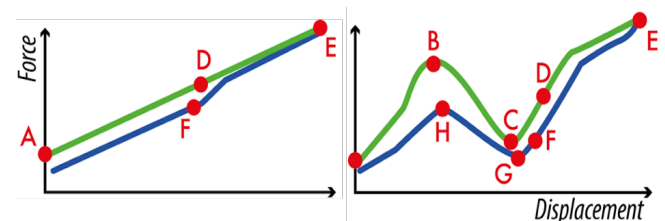
## PRELIMINARIES: DESIGN PARAMETERS OF BUTTONS

Here we review key design parameters of three common types of buttons: physical, touch, and mid-air. This serves as background for mechanical modeling of buttons.

*Physical buttons* are electromechanical devices that make or break a signal when pushed, then return to initial (or re-pushable) state when released. Physical dimensions (width, slant, and key depth), materials (e.g., plastics), and system-level feedback (modalities and latencies of feedback) are well-known design parameters [46]. The travel distance at which the button is activated is called its *activation point* (see [67]).

A central design consideration is the *force–displacement function* or force curve [20, 46, 51, 58]. Figure 2 shows stereotyped FD curves for linear and tactile push-buttons. We denote displacement at a point as  $D(X)$ , force at a point as  $F(X)$ , displacement difference between points as  $dD(X - Y)$ , and force difference between points as  $dF(X - Y)$ . *Linear buttons* have no tactile landmarks during press-down. The button has the feel of pressing a simple spring.  $D(E)$  is called full-travel distance and  $D(D)$  is known as make-travel or activation-travel, while  $dD(D - E)$  is called over-travel.  $dD(D - F)$  is called hysteresis. Hysteresis is designed to prevent unintentional double-clicking. The main engineering parameter is the position of *make*  $D(D)$  and *break* points  $F(F)$ . On the other hand, *tactile* and “*clicky*” buttons offer more points of interest (POIs), or changes during press-down and release.  $F(B)$  is called actuation force, and  $dF(B - C)/F(B)$  is called snap ratio and determines the intensity of tactile “bump.” A snap ratio greater than 40% is recommended for a strong tactile feeling by rubber-dome manufacturers. With some tactile buttons, a distinct audible “click” sound may be generated, often near the *snap* or *make* points.  $F(G)$  is release force, and the amount of  $dF(H - G)$  determines the intensity of snap feeling when releasing a button.

From the force curve point of view, *touch buttons* are zero-travel buttons. Consequently, they show lower peak force than physical buttons do. Because of false activations, the finger cannot rest on the surface. Activation is triggered by thresholding the contact area of the pulp of the finger on the surface. *Mid-air buttons* are based not on electromechanical sensing but, for example, on computer vision or electromyographic sensing. Since they are contactless, they do not have a force curve. The activation point is determined by reference to angle at joint or distance traveled by the fingertip. Latency and inaccuracies in tracking are known issues.



**Figure 2.** Idealized force–displacement curves for linear (left) and tactile (right) buttons. Green lines are press and blue lines are release curves. Annotations (A–H) are covered in the text.

## THEORETICAL ASSUMPTIONS AND SCOPE

**Neuromechanics** proceeds from the biological fact that the central nervous system has no direct access to the physical body (here, finger) it is moving. It is enveloped by neural interfaces. This idea is consistent with information-theoretical views of a capacity-limited motor system [63]. Capacity limits predict bounds to accuracy and time costs of human responses. However, as do control-theoretical models [30, 50], neuromechanics also subscribes to modeling of essential features of the plant, the body or pointer in its environment, as well as their properties such as sampling rate, latency, and gain. As in biomechanics [6], the plant model in neuromechanics captures essential anatomical and physiological factors, the bones and tissues of the fingertip in our case, as well as the mechanical properties of the button they contact. Combining these assumptions, one can study how the motor system achieves control over episodes with a button. However, this general frame could be instantiated in many ways. We here present theoretical assumptions and scope behind NEUROMECHANIC.

**PERCEPTUAL CONTROL** Because of the enveloping assumption, it follows that users cannot control or perceive events *in* the button. They can only perceive feedback signals produced during and after button-pressing. Moreover, while a button can be explored in a closed-loop fashion, regular keypresses are too rapid for online corrections. We therefore approach button-pressing as an episodic open-loop perceptual control problem [8, 11, 19, 55, 56, 57]. We hypothesize that it is about minimizing discrepancy between expected and perceived sensations of the button being activated. Users intend to cause a perception of the button being activated at some moment in time, such as when a character is perceived to jump over a platform in a game or an LED lights up in calling an elevator, or the sensation of the finger impacting the “clicky” region of a tactile button. To improve motor commands, they compare their expected and obtained sensations of button activation timing. Figure 3 illustrates the idea. We formulate an objective function capturing these ideas.

**PROBABILISTIC INTERNAL MODEL** Because button-pressing is an episodic open-loop problem, the motor system, in order to adapt to a new button, must learn how its actions are mapped to observable results. An internal model is needed [35]. Probabilistic internal models are ideal, since they cope with noise and uncertainty [40, 62]. They posit an updateable probability distribution that maps motor commands with sensations. From this viewpoint, each attempt to press a button is a “gamble.” The distribution is updated after observing of the result. Because of the nonlinear nature of these observations, the internal model needs a matching level of representational power. We use a Gaussian process model for this end.

**NOISY NEURAL TRANSMISSION AND MUSCLE ACTIVATION** The internal model cannot control the finger directly, only activation potentials sent to muscles. However, the neurally transmitted signal, or innervation, is compromised by noise, loss, and delay. Moreover, the muscle, itself nonlinearly responding, is susceptible to noise. Because of these two sources of noise, finger movements are stochastic. We use Gaussian additive noise as a model of some basic effects of neuromuscular noise.

**BIOMECHANICS** The command to activate the muscles triggers a dynamic cascade of physical events. In button-pressing the physical objects involved are muscles, the finger bone, the pulp of the fingertip, the cap of the button, its internal mechanism (e.g., spring), and the ground. We model the muscle by using a well-known Hill-type model. It attaches to a rigid object (bone) with a compressing pulp consisting of a series of springs. Different button designs with travel depths, activation points, and force curves can now be physically modeled. Although movement dynamics are nonlinear, they can be simulated accurately via physical simulation.

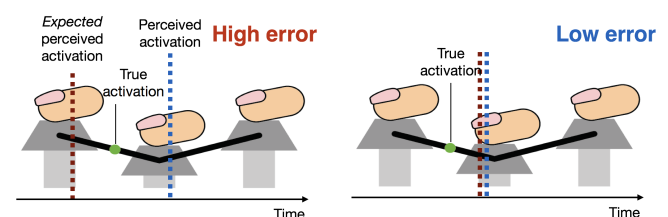
**MULTIPLE NOISY SENSORY SIGNALS** A button press produces several sensory signals. We argue that the key feedbacks in button-pressing are mechanoreceptive (tactile sensation), proprioceptive (joint angles), auditory (‘beep’), and visual signals (event on display). We further assume that signals are susceptible to modality-specific Gaussian additive noise. We use differences in signal properties to explain why, for example, mid-air buttons perform worse than tactile buttons.

**CUE INTEGRATION** After a keypress, the motor system has several sensory noisy feedbacks available. It needs to construct an integrated perception of “what happened” in order to tune the motor command. We assume that it uses sensory signals optimally to infer the *moment* when the button was activated. This estimate is used by the motor controller to achieve increasingly better (temporal) control of button activation.

We use *cue integration theory* to compute an integrated p-center [17, 18, 40]. It assumes that perception is a probabilistic process that takes the variance of a sensory signal into account when judging it in the context of other signals. This way, the motor system can “decode” the timing of button activation, or p-center, even if feedback information is compromised.

## Modeling Objectives and Scope

Our goal is a modeling tool for the study and engineering of buttons. Therefore, in implementing these ideas, we aim for neither a theory with minimal assumptions nor one with highest biological plausibility but one that yields realistic predictions across a range of common designs. We strive to avoid free parameters that lack theoretical content or are empirically unidentifiable. A researcher should be able to model button designs and tasks by changing physically/anatomically or otherwise determinate parameters. We also confine this work to the simple press and leave aiming for future work.



**Figure 3.** Perceptual control of a button: the motor system has no access to the true moment when the button is activated, but it can try to reduce *error* (distance) between estimated and perceived sensations. Left: perceptual control fails when error is high. Right: precise control is achieved when estimated and perceived activation co-occur in time.



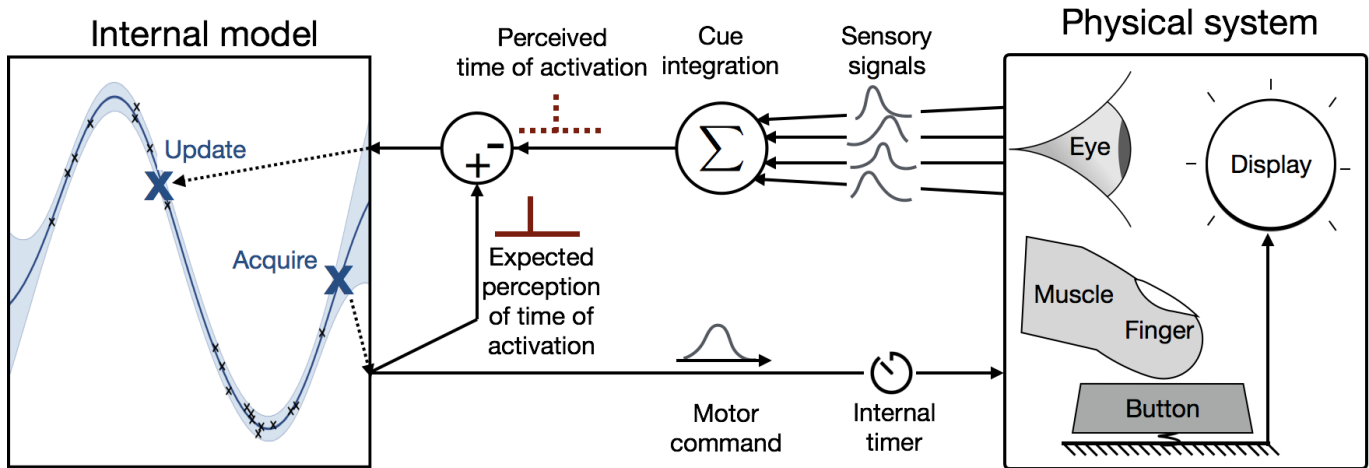


Figure 4. NEUROMECHANIC is a computational model of neuromechanics in button-pressing. It implements a probabilistic internal model (Gaussian process regression) that attempts to minimize error between its expected and perceived button activation. Its motor commands are transferred via a noisy and delayed neural channel to muscles controlling the finger. A physical simulation of the finger acting on the button yields four types of sensory signals that are integrated into a single percept (p-center) by means of cue integration (implemented via a maximum likelihood estimator).

### NEUROMECHANIC: A COMPUTATIONAL MODEL

NEUROMECHANIC implements these ideas computationally. It consists of two connected sub-models (Figure 4).

**The physical system simulator** takes as input a continuous activation signal sent to agonist and antagonist muscles. The muscles act on a model of the bone moving up/down. The mechanics of the three main types of button design mentioned above can be modeled. The physical simulation yields continuous kinematics, dynamics, and sensory signals for proprioception (fingertip displacement), tactition (pulp compression), vision (light on display), and audition (beep sound).

**The motor controller** is a Bayesian optimizer using a Gaussian process (GP) regression model initiated with a uniform prior. The model maps motor command parameters to error in its perceptual estimate. It is used to compute an activation command sent to the finger muscles. We also discuss how to compute an expectation for the sensations that result from a button press. We call that the integrated perceptual center (p-center). To optimize movements, *expected* p-center is compared against the *observed* p-center. P-center is computed from multiple sensory feedback signals by means of cue integration. After every observation, the internal model (GP) is updated to sample a better motor command.

### Operation and Outputs

Operation of the model is split into two parts: training and testing. In *training*, the motor control model learns to use the button over a number of learning episodes. Repeated attempts to activate the button (here, 10) are given in one learning trial to estimate the objective score. After training, the best command is *tested* repeatedly to obtain statistical distributions for output variables. Dynamics and kinematics can be investigated, such as velocities and pulp compressions over time. Task-level metrics include (1) the probability of activating the button and (2) indices of perceptual accuracy and precision. Perceptual precision can be used in estimating error rates in tasks that require temporal precision.

### Objective Function

A motor command  $\theta$  sent to the finger muscles consists of three parameters:

$$\theta = \{\mu_{A+}, \tau_{A+}, \sigma_{A+}\} \quad (1)$$

with signal offset  $\mu$ , signal amplitude  $\tau$ , and duration  $\sigma$  of the agonist (A+) muscle. We have set physiologically plausible extrema (min and max) for the activation parameters. Note that this formulation assumes that the antagonist muscle resists motion passively. More determinate pull-up motion can be achieved by adding similar parameters for the antagonist muscle (A-).

The objective is to determine motor command ( $\theta$ ) and associated estimate of perceived button activation ( $pc_e$ ) that minimize error:

$$\min_{\theta, pc_e} E_P(\theta, pc_e) + E_A(\theta) + E_C(\theta) + \omega F_M(\theta) \quad (2)$$

where  $E_P$  is perceptual error,  $E_A$  is error in activating the button, and  $E_C$  is error in making contact (button cap not touched).  $F_M$  is muscle force expenditure computed from the Hill muscle model (see below), and  $\omega$  is a tuning factor. We assume that activation and contact errors are trivial to perceive. Therefore,  $E_A$  and  $E_C$  are binary: 1 in the case of error and 0 otherwise. Perceptual error  $E_P$  is defined as distance (in time) between expected p-center  $pc_e$  and observed p-center  $pc_o$ :

$$E_P = |pc_e - pc_o| \quad (3)$$

The computation of  $pc_o$  is given below. To prevent NEUROMECHANIC pushing the button with unrealistically high force, which would in reality cause fatigue and stress, we introduced a controllable ergonomics (or effort) term to the objective ( $F_M$ ). By changing its weight ( $\omega$ ), the model can simulate, for example, a user trading off effort versus temporal precision, or a user not caring about temporal precision or force but only about activating the button.

### Internal Model

The motor controller is a Bayesian optimizer (BO). It decides variables  $pc_e$  and  $\theta$  in order to optimize the perceptual objective given above. For the BO, the motor command is a random function. In our implementation, a Gaussian process (GP) regression model is the internal model that maps  $\theta$  and  $pc_e$  with the objective score. Observations are assumed to have added Gaussian noise. The prior's joint distribution of the function values is multivariate normal. In the GP model, objective score is estimated for a selected grid, and a point is selected where error is minimal. The BO's *acquisition function* controls sampling of the next motor command from the model. Favoring exploitation leads to quick convergence to some acceptable motor command. Favoring exploration slows convergence but has a better chance of finding the globally optimal motor command. In this way, the model implements what is called *active inference* [64]: the idea that motor learning involves actively trying out of motor hypotheses.

### Neural Transmission: Noise and Delay

The motor command sent by the central nervous system is corrupted by additive Gaussian noise with noise parameter  $\sigma_n$  (see Table 1). Note that in our model this represents the additive contribution of neural and muscular noise. Since the forces in button-pressing are relatively small, for simplicity, we do not assume noise to be signal-dependent as in gross movements. This noise-corrupted signal must be sent to the muscles at a desired point in time. However, the onset of the signal cannot be decided at arbitrary precision. We use the Treisman internal clock model [70] for gating the onset of  $\mu_{A+}$ . Treisman proposed a pacemaker-plus-counter-plus-comparison model. The pacemaker produces a series of pulses at 70 ms intervals. Although this parameter is person-specific, we use a population-level estimate in the model.

### Finger Bone and Pulp

Although finger density, length, and size can vary, a finger can be regarded as a cylinder-shaped object with some density (see Figure 5). At the center of the finger is a cylinder-shaped bone, and the soft finger pulp surrounds it. The bone is assumed to be a rigid body, and the soft characteristics of the finger pulp are approximated as a spring-damper system of stiffness value  $k_f$  and damping coefficient  $c_f$ . The physical contact between the finger and the button is understood as the dynamics of the connection of two mass-spring-damper systems in series.

### Muscles

The force to move the fingers comes from the contraction of an agonist muscle (A+) and an antagonist muscle (A-) resisting it. The parameters of the motor command  $\theta$  represent the potentiating signal sent to the agonist muscle. The original signal is a rectangular pulse with a specific duration and width. This signal is then low-pass filtered to simulate motor neurons' intrinsic dynamics [28] (resonant frequency around 4 Hz). This yields a plausible shape for the potentiating signal.

The two muscles are arranged in the longitudinal direction of the finger, causing the fingers to rotate around the near joint. We approximated the situation in which both muscles exert a force on the finger in a direction perpendicular to the ground

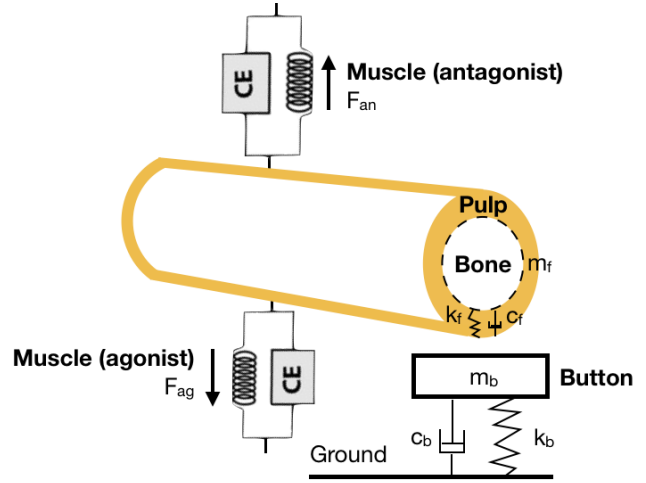


Figure 5. Overview of finger and button models.

(Figure 5). The antagonist muscle exerts a force to pull the finger upward, and the agonist muscle pulls it downward.

We model the dynamics of the muscles via the well-known Hill-type model [24, 65]. This model type approximates the muscles with one serial spring and a parallel spring connected to one contractile element (CE). Of these, the spring connected in series to the CE represents the tendon that is stretched when the muscle contracts and produces the restoring force. The CE contracts on the basis of an activation signal (between 0 and 1) coming from the motor neurons. When the CE contracts, the spring in parallel with the CE too shortens, creating a passive restoring force in the direction opposite the muscle contraction. When the agonist muscle contracts, the finger moves down and thus stretches the parallel spring of the antagonist muscle, creating a restoring force in the upward direction. However, this kind of coupling between the antagonist and agonist muscle was excluded from our finger model. This is a reasonable assumption because there is little movement to the extent that muscles are actually stretched in real life [45]. Accordingly, we express the force from a muscle acting on a finger as the sum of the restoring force  $f_p$  caused by the contraction of the parallel spring and the contracting force  $f_c$  produced by the CE:  $f = f_c + f_p$ . These two forces always act in opposite directions. More specifically,  $f_p$  can be expressed as a force from an exponential spring [45]:

$$f_p = \max(0, k_s(\exp(k_c e) - 1) + k_d \dot{e}) \quad (4)$$

Here, the max function is implemented to exclude the spring restoration force that occurs when the muscle is stretched.  $k_s$ ,  $k_c$ , and  $k_d$  characterize the stiffness and damping of the element.  $e$  is the strain value of the current muscle relative to the initial length of the muscle, and  $\dot{e}$  is the first time derivative of the strain. Under the Hill-type model,  $f_c$  is expressed as the product of activation signal  $a$ , force-length relationship  $F_l(l)$ , and force-velocity relationship  $F_v(\dot{l})$ :

$$f_c = a F_l(l) F_v(\dot{l}) \quad (5)$$

Here,  $l$  is the absolute length of the muscle and  $\dot{l}$  is the time derivative of muscle length. We adopted the linearized  $F_v$ ,  $F_l$  functions used from [44, 45].

## Button

Our button physics model is straightforward but can represent many types of buttons. First, we assume that the button is activated only for user movement in a certain direction. This assumption can be implemented by representing the physical state of the button as a one-dimensional variable,  $x_b$ . For a physical push-button,  $x_b$  is the vertical distance of the button from the fixed ground. For touchscreen-based software buttons,  $x_b$  could be the contact area of the finger and the screen. Next, the physical body of the button was modeled as a one-dimensional translational mass-spring-damper system, which is coupled to an immobile rigid ground (see Figure 5).

This is a generic model that can describe the physical characteristics of various buttons. A touchscreen with little deformation when pressed by the user can be considered as a mass-spring-damper system with a very high stiffness value ( $k_b$ ) and a large moving mass ( $m_b$ ) in this model. A typical physical button has a relatively light moving mass and a low stiffness value with easy deformation.

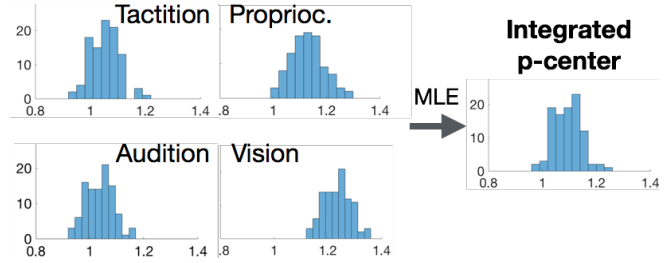
## Display

The computer converts electrical signal  $x_b$  into a discrete event, the activation point. This is modeled with an algorithm that triggers an event when  $x_b$  exceeds a threshold value. This event causes two feedback signals: visual and auditory (beep). The event-to-display and event-to-beep delays are assumed to be Gaussian random variables.

## Sensory Feedback and Computation of p-Centers

The finger-button-display system is connected to four sensors: mechanoreception, proprioception, audition, and vision. Each sensory modality  $i$  produces a p-center  $pc_i$ .

Mechanotransduction is the transfer of a neural signal evoked by physical stimulation of mechanoreceptors. We are especially interested in how mechanoreceptors on the finger pad convey information about the timing of a button press. Slowly adapting (SA1) fiber types are sensitive to coarse spatial structure (here, the sensation of the flat top surface of the button), and rapidly adapting (RA) fibers respond to motion. Kim et al. [38] suggest that the main signals from the fingertip are position, velocity, acceleration, and jerk from the finger pulp. However, because contact force and indentation have a nearly linear relationship, and force correlates highly with acceleration at button impact, and because buttons have little localized information, we limited our model to mechanoreception caused by pulp compression. Its time-varying signal is rectified and split into negative and positive components. In pulp compression, these correspond to indentation and retraction of the fingertip. The full model of Kim et al. includes consideration of the relationship between spiking and the depolarization of membrane potential with refractory periods. However, since button-pressing is a short event with a single maximum for indentation and retraction, the peak of those signals offers the most reliable point in the psychophysiological event [49]. We represent each rectified time-varying signal with the offset of its peak, the p-center. This peak is identified as the maximum of sensory signal *after* signal-specific Gaussian noise ( $\sigma_m$ ) is applied (see Table 1).



**Figure 6.** NEUROMECHANIC computes an estimate of *when* the button was activated from noisy, continuous sensory signals. It detects the peak of each signal (modality-specific p-center) and uses MLE to compute the integrated p-center. Data from simulations with a tactile button.

Proprioception is similarly modeled as the peak of the noise-corrupted ( $\sigma_p$ ) signal caused by the displacement of the finger. Audition registers a single event (onset of beep) after latency ( $\sigma_a$ ). Similarly for vision, a single event (onset of display) is registered after latency ( $\sigma_v$ ).

## Probabilistic Cue Integration with MLE

Each p-center  $pc_i$  conveys a cue: the moment in time of peak sensation in that sensory signal. However, the signals are unreliable to varying degrees. The integrated center  $pc_o$  represents a sensation integrated over all noisy signals. In our model, it describes a timing estimate: when the button was pressed. We use maximum likelihood estimation (MLE) to obtain a point estimate of  $pc_o$ . For another implementation of cue integration, see [40]. In MLE, assuming that a single-cue estimate is unbiased but corrupted by Gaussian noise, the optimal strategy for estimating  $pc_o$  is a weighted average [17, 18]:

$$pc_o = \sum_i w_i pc_i \text{ where } w_i = \frac{1/\sigma_i^2}{\sum_i 1/\sigma_i^2} \quad (6)$$

with  $w_i$  being the weight given to the  $i$ th single-cue estimate and  $\sigma_i^2$  being that estimate's variance. Figure 6 shows exemplary p-center calculations: signal-specific ( $pc_i$ ) and integrated p-centers ( $pc_o$ ) from 100 simulated runs of NEUROMECHANIC pressing a tactile button. Note that absolute differences among  $pc_i$  do not affect  $pc_o$ ; only signal variances do. The integrated timing estimate is assumed to be robust to long delays in, say, auditory or visual feedback. This assumption is based on a study showing that physiological events that take place quickly, within a few hundred milliseconds, do not tend to be cause over- or underestimations of event durations [15].

## IMPLEMENTATION AND PARAMETER SELECTION

NEUROMECHANIC is implemented in MATLAB, using BAYESOPT for Bayesian optimization (the GP model uses the ARD Matern 5/2 kernel), SIMSCAPE for mechanics, and SIMULINK for the muscle model.

## Parameter-Setting

Model parameters are listed in Table 1 with values and sources. These consist of physical, physiological, and neural parameters. *Physical parameters* are straightforward, because they represent physically measurable aspects of buttons or fingers. A designer or engineer may change them to model different buttons or fingers. *Physiological parameters* describe muscle properties and are adopted from biomechanics literature.



**Table 1. Model parameters.** Button parameters here given for physical buttons, while task parameters (e.g., finger starting height) are given in text;  $f$  denotes function

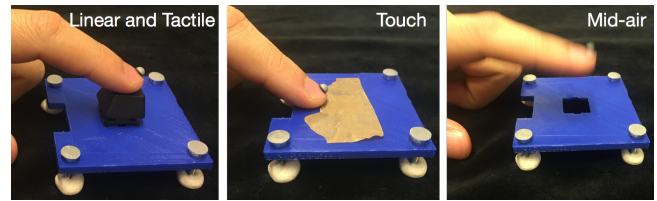
Variable	Description	Value, Unit	Ref.
$f_r$	Radius of finger cone	7.0 mm	
$f_w$	Length of finger	60 mm	
$\rho_f$	Density of finger	985 kg/m <sup>3</sup>	
$c_f$	Damping of finger pulp	1.5 N·s/m	[72]
$k_f$	Stiffness of finger pulp	$f$ , N/m	[73]
$w_b$	Width of key cap	14 mm	
$d_b$	Depth of key cap	10 mm	
$\rho_b$	Density of key cap	700 kg/m <sup>3</sup>	
$c_b$	Damping of button	0.1 N·s/m	
$k_s$	Elasticity of muscle	0.8·PCSA	[44]
$k_d$	Elasticity of muscle	0.1· $k_s$	[44]
$k_c$	Damping of muscle	6 N·s/m	[44]
PCSA	Phys. cross-sectional area	4 cm <sup>2</sup>	
$L_{0ag}$ , $L_{0an}$	Initial muscle length	300 mm	
$\sigma_n$	Neuromuscular noise	$5 \cdot 10^{-2}$	
$\sigma_m$	Mechanoreception noise	$1 \cdot 10^{-8}$	
$\sigma_p$	Proprioception noise	$8 \cdot 10^{-7}$	
$\sigma_a$	Sound and audition noise	$5 \cdot 10^{-4}$	
$\sigma_v$	Display and vision noise	$2 \cdot 10^{-2}$	

*Noise parameters*, by contrast, cannot be taken from literature, because they are particular to this neuromechanics system. We have set these parameters manually in order to reproduce certain basic effects: Neuromuscular noise, which reflects the joint additive contribution of neural and muscular noise, was set manually to a low, conservative value. Noise values for mechanoreception and proprioception were set to reflect the finding that proprioceptive signal has less resolution and slower response than tactile sensation [31]. Parameters for audition and vision are jointly determined by the latency of the computer mediating the feedback signals *and* the sensory systems. The former can be measured [12], but the latter has to be estimated. In this paper, we have set them to correspond to a finding according to which the auditory modality is faster and more reliable than the visual [3]. Noise is lower for audition also because auditory feedback can be produced by both the button (“click”) and the computer.

### Bayesian Optimization

The system includes also hyper-parameters of the BO. The exploration ratio was set to 0.2. We use expected improvement (EI) for the acquisition function in BAYESOPT. It tries to maximize the expected improvement over the current best. In our simulations, unless otherwise stated, we set the ergonomics (effort) factor  $\omega$  to 0.33.

**TRAINING AND TESTING** In all simulations reported on in this paper, total learning time was set to 10 minutes (wall-clock time on a Macbook Pro 2.8 GHz i7). Each trial includes 10 button-press attempts repeated with the same motor command. Objective score is computed as an average over all 10 attempts. After training, the best-known model is chosen, and 100 repeated independent press attempts are provided for the button. Final measurements are computed from the data from these.



**Figure 7. Data collection on kinematics.** High-fidelity optical motion tracking was used to track a marker on the finger nail. A custom-made single-button setup was created, using switches and key caps from commercial keyboards.

### SIMULATING BUTTON PRESSES

We assessed the validity of NEUROMECHANIC’s predictions in a series of simulations testing four button types: tactile, linear, touch, and mid-air. Comparing the tactile and linear type is challenging, because the only difference is the slight tactile “bump.” Touch buttons are generally considered worse than physical buttons, but theoretical explanations are lacking here. We are also interested in what the model predicts for mid-air buttons, which lack mechanoreceptive feedback entirely and rely on proprioceptive feedback. Besides the physically measured parameters of buttons, and objective weight (for “effort,”  $\omega$ ), used to explore users’ button-pressing strategies, all parameters were set as described above.

We assess simulation predictions against empirical evidence in two ways: (1) quantitatively, against kinematics profiles of a human participant, and (2) qualitatively, against statistically significant main effects reported in prior empirical literature. We inspect predictions for displacement–velocity curves, force–displacement curves, muscle forces, and task-level measures (perceptual error and button activation timing).

### Button Modeling

To obtain realistic force–displacement (FD) curves for the simulation, we measured several mechanical buttons with a custom-made instrument. A stepper motor rotates a precision lead screw to move a stage. The stage has a load cell to press a button. It yields a list of displacement–force pairs (at 20  $\mu$ m resolution) obtained during pressing and releasing a button. The apparatus is similar to [51]. Four buttons were modeled:

- *Linear push-button*: Force curve adopted from a Cherry MX Black switch.
- *Tactile push-button*: Force curve adopted from a Cherry MX Blue switch.
- *Touch button*: Activation at contact with surface.
- *Mid-air button*: Contactless activation at 3 cm below finger starting height.

In the case of tactile and linear buttons, the finger was assumed to start on top of the key (1 mm SD). In NEUROMECHANIC, in the case of touch and mid-air buttons, finger starting height was set at 3 cm from surface (SD 7 mm) following an earlier paper [27].

### Kinematics Dataset: A Single-Subject Study

To enable comparing simulated kinematics to human performance for the four buttons, motion tracking data on finger kinematics were collected from a single subject (co-author

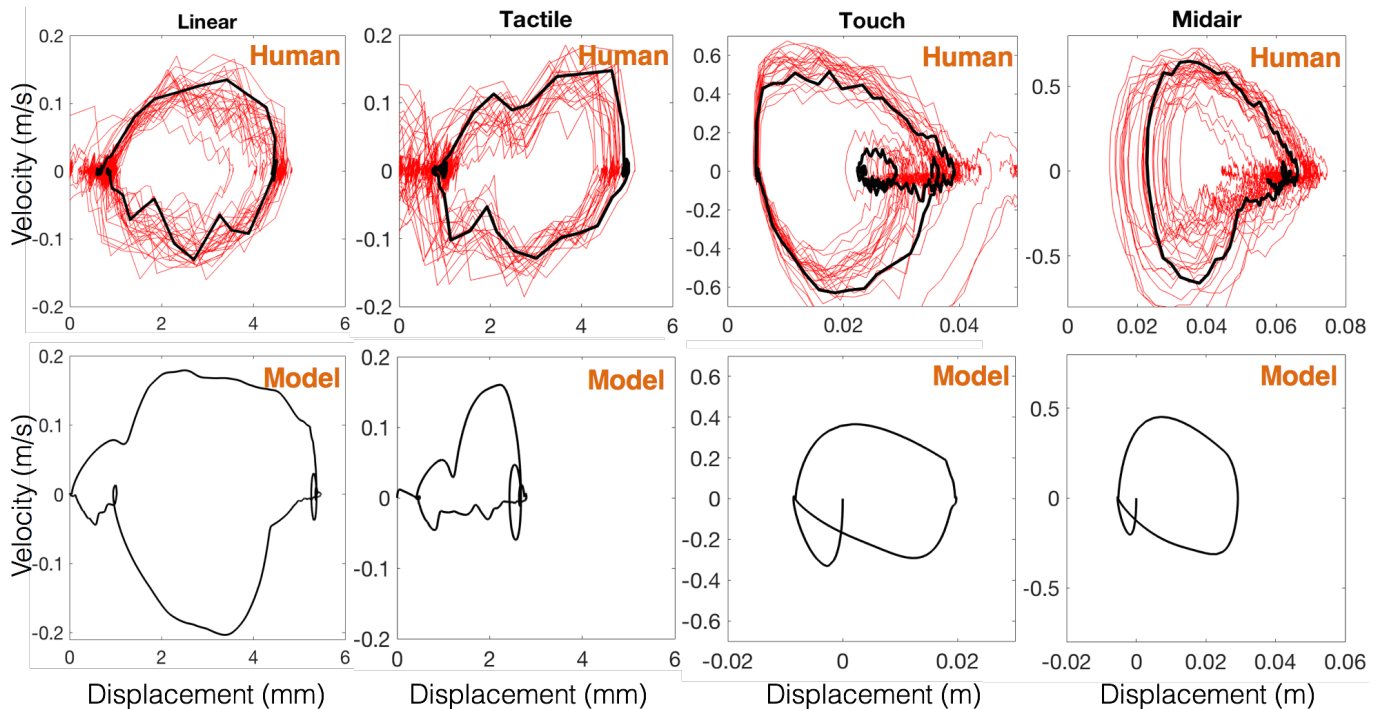


Figure 8. Displacement–velocity curves for four button types from single-subject recordings (top) and simulations (bottom).

SK). The user was asked to press the button rhythmically, finding a “natural” way to press. Motion data for 10–20 presses per button were collected after sufficient practice. The position and motion of the finger were tracked with a high-rate motion tracking system: an OPTITRACK system, consisting of eight cameras, recording at 240 fps and organized for a desktop setup. A reflective adhesive marker was positioned in the middle of the index finger’s nail. A custom-made single-button apparatus was used. The push-button designs and the task conditions (finger height) were matched with the parameters given above. Figure 7 shows the data collection setup.

### Results: Kinematics

Displacement–velocity (DV) curves for the four buttons are shown separately for human (top) and simulated data (bottom) in Figure 8. We here inspect key features of the DV curves.

**VELOCITY RANGES** Predicted velocity ranges (on the y-axis) match the data reasonably well, especially given that no model parameters were fitted to the data. A few exceptions are clear. The release part of the DV curve for the tactile button shows lower velocity in our model. The touch button is another exception. The participant showed higher peak velocities than the model. We hypothesized that the participant might have used much higher force in cases such as these two than in the others. We tested this hypothesis by simulating a user with low ergonomics objective. We decreased the ergonomics (effort) objective,  $\omega$ , from 0.33 to 0.20. Figure 9 shows DV curves obtained for this modified user. The peak velocity of 0.45 m/s matches much better with the participant data. We conclude that the force used is a strategic factor affecting kinematics and can be captured reasonably well by changing the weight of the ergonomics objective of NEUROMECHANIC.

**DV CURVE PROFILES** The main characteristics of the DV curves are reproduced reasonably well. We discuss each of the four button types in turn. First, in the linear button condition, both the participant and the model exhibit a ballistic compress (push-down) motion. However, the model fails to predict as high velocity upon release as the data show. Second, Figure 10 shows annotated features for the tactile button. The tactile bump is clearly visible – in compress (push-down) and release (pull-up) phase both. However, the release phase predicted by the model shows lower velocities. Third, in the touch button condition, the overall shape matches reasonably well. However, the participant exhibited a strategy that the model did not predict (Figure 8): the user kept the finger down after hitting the surface, causing visible jitter. This behavior is not present in our idealized model behavior that pulls the finger up immediately. Fourth, in the mid-air condition, we see high compress-phase velocities for both model and user data. However, again, the participant tended to drag the finger for a time while it was pushed down before pulling it up, whereas the model pulls up immediately. The model’s immediate pull-up is caused by the antagonist muscle force. We note also that the asymmetric, elliptical shape predicted for the mid-air button closely replicates the FD curve reported in an earlier empirical study of mid-air kinematics (Figure 2 in [69]).

### Results: Success Rate and Temporal Precision

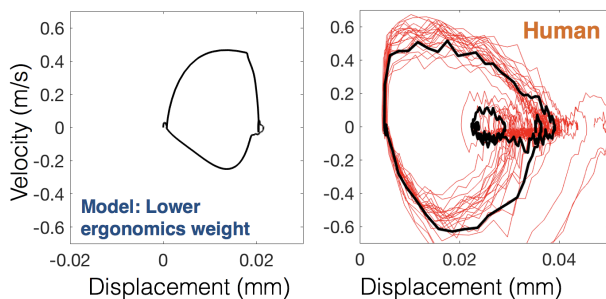
Simulation results for perceptual error, temporal precision (SD of perceptual error), variability (SD) of button activation timing, and success rate in button activation are given in Table 2. Since our empirical dataset does not include measures for success or for precision, we compare model predictions to main effects reported in the literature.



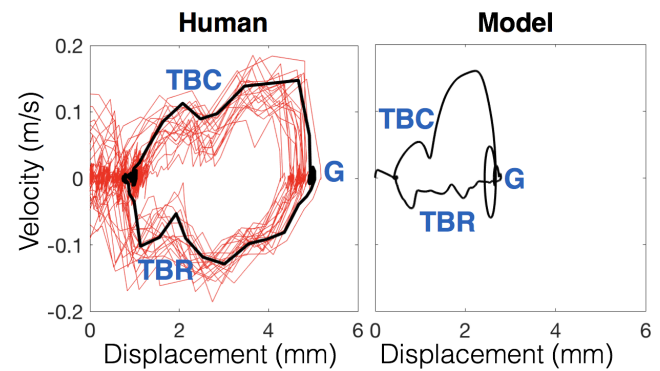
**TOUCH VS. TACTILE BUTTONS** The lowest perceptual error is predicted for the touch button (avg. 34 ms). The tactile button is slightly worse (40 ms). However, the standard deviation of perceptual error is much higher for the touch button, at 76 ms, while only 26 ms for the tactile button. This prediction is consistent with a recent empirical finding from a temporal pointing task [42]. In temporal pointing, a button must be activated within some limited but predictable time window. This task is common in music and gaming applications. The authors found touch buttons to yield 4.5% worse error rates in this task. Although the quantitative difference we predict is much larger (34%), the direction of the effect is the same. That touch buttons are inferior in tasks requiring temporal movement control is in line also with other findings. For example, touch buttons have been found to be inferior in typing [14, 53]. In NEUROMECHANIC, these differences are partially due to higher variation in the finger starting height: the possibility of keeping the finger on top of the physical button eliminates one source of timing uncertainty.

**LINEAR VS. TACTILE BUTTONS** The linear button was predicted to be slightly worse than the tactile, with 47 ms error and 31 ms standard deviation. This difference was predicted by an early study showing that a tactile switch performs better than a linear one in terms of speed and error [9]. While a later study failed to replicate this effect [1], there is other evidence suggesting that the tactile design may be superior. For example, users are known to prefer FD curves that are “roller-coaster-shaped” – i.e., closer to the tactile type [46]. In NEUROMECHANIC, more work is needed to explain this. We hypothesize that the difference may be attributable to the tactile bump creating a “secondary” p-center.

**MID-AIR BUTTONS** The mid-air button showed the worst performance of all four button types, with very high perceptual error: 178 ms (SD 47). While hard data on temporal precision could not be found for mid-air buttons, their generally low performance is well-reported [47, 48, 66]. The model predicts that users of mid-air buttons have difficulties especially in estimating when they activate the button. This is because the mechanoreceptive signal is missing. The user relies on proprioception and (degraded) auditory and visual feedback for timing estimates. We found that in the model, this is quite visible in a wider p-center distribution. Most p-centers occur within a 150 ms range, showing uncertainty of the perceptual system. In contrast, p-centers for the other button types fall



**Figure 9.** Displacement–velocity curves for the touch button after lowering of ergonomics weight ( $\omega$ ) in the objective function. Higher velocities emerge than in Figure 8, yielding a better fit.



**Figure 10.** Features of displacement–velocity curves (tactile button) in human and model data: TBC = Tactile bump during button compress phase; TBR = Tactile bump during release; G = Button ground.

within a 100 ms range. Finally, while other buttons had high or near-perfect activation success, the mid-air button showed success only 54% of the time. We learned that this is because of the higher forces involved. NEUROMECHANIC obtains larger rewards in the objective score by minimizing the use of force than by minimizing  $E_p$ . However, by setting a lower value to  $\omega$  (ergonomics weight), we can drive the model to a near-perfect success rate. Perceptual error is increased, though.

### Results: Muscle Forces

Table 2 shows peak muscle forces for the four buttons. We note that, overall, the absolute values are in line with prior literature on push-buttons. Forces reported for tactile buttons are between 1.7 and 2.0 N for 0.6 N make-force keyswitches [36]. Our prediction was 1.4 N for the tactile and 1.6 N for the linear type. The model also reproduces the effect of peak forces being sensitive to the shape of the FD curve [60].

**ADAPTING USE OF FORCE TO THE TASK** It is well-known that users adapt their force-use to the task. For instance, in tasks that require temporal precision, users are likely to press harder to obtain stronger mechanoreceptive stimulations [2]. In a study of tapping with a touch button [29], participants pressed a button with 5–7 N of force even if the required activation force was much lower (0.98 N). In NEUROMECHANIC, increasing impact force improves timing estimates (p-centers) and thereby temporal precision of movements. The tradeoff between force-use and temporal precision in the objective function is controlled by the tuning factor  $\omega$ . When  $\omega$  is zero, the peak muscle force for a tactile button increases to 2.45 N. The FD responses change visibly. With a slight increase in  $\omega$ , peak muscle force drops to 2.0 N. We conclude that the model can meaningfully adapt its use of force, with simulation predictions ranging from 1.4 N to 2.4 N.

**Table 2.** Simulation results for the four button types

	Linear	Tactile	Touch	Mid-air
Perceptual error	47 ms	40 ms	34 ms	178 ms
SD of perc. error	31 ms	26 ms	76 ms	47 ms
SD of activation time	52 ms	43 ms	90 ms	51 ms
Activation success	92%	82%	94%	54%
Peak muscle force	1.65 N	1.41 N	2.6 N	2.9 N

## DISCUSSION

From the vantage point of the central nervous system, input devices are black boxes. As W. R. Ashby wrote in *Introduction to Cybernetics*, “The child who tries to open the door has to manipulate the handle so as to produce the desired movement at the latch; and he has to learn to control the one by the other without seeing the internal mechanism that links them. [...] We are confronted at every turn with systems whose internal mechanisms are not fully open to inspection” [5, p. 86]. Neuromechanics can illuminate this fundamental aspect of interaction. Neuromechanics integrates insights from decades of research into a unifying framework. It assumes a noise-limited channel as information-theoretical accounts of performance do. However, unlike, for example, Fitts’ law, it predicts the dynamic consequences of that noise downstream. As do control-theoretical models of pointing, which also model dynamics, neuromechanics views interaction as a continuous problem. However, control-theoretical models and optimal control models published in HCI thus far have lacked biologically plausible accounts of the controller and the plant. This has complicated applications in design, because model parameters must be empirically obtained for every condition [50]. Biomechanical models do offer detailed descriptions of the plant, but without an account of the controller, they have limited use beyond descriptions of observed data.

Neuromechanics views button-pressing as optimal control limited by physical, neural, and sensory bounds. In our study, we defined the motor control problem, proposed theoretical ideas on how the motor system solves them, implemented them computationally, and inspected predictions against empirical data. We sum up the findings thus: The model can – even with minimal fitting of parameters to data – predict some core characteristics of FD and DV patterns, as well as the use of force and the temporal precision for four distinct button types. The model fared less well in two respects: first, in failing to predict the ranges of kinematics variables for some buttons and, second, in failing to predict varied button-pressing strategies. Although much work remains to be done, the results are promising and support the optimality assumption.

We conclude that neuromechanics deserves more attention as a rigorous and biologically motivated theory for the study of input methods.

## FUTURE WORK

This paper is a starting point for deeper investigation of button-pressing. Neuromechanics offers a simulation platform that allows theorists to critically test their hypotheses and designers to test their designs. The model already allows many more analyses to be carried out for better understanding of button-pressing, as in looking at the effects of feedback design or of walking (when the finger is oscillating) or at effects of impairments such as essential tremor. But we see four major opportunities to extend the validity of the model’s predictions and the scope of modeled buttons.

**NOISE** Modeling latent neural and cognitive constructs, such as noise, poses a scientific challenge for future research. Change in noise parameters has a large and poorly understood effect on dynamics downstream. However, without noise, a button can

be activated with arbitrary precision. For example, lowering sensory noise parameters to  $10^{-9}$  reduces perceptual error to the order of  $1.5 \cdot 10^{-6}$  s. Our noise model was tuned manually to reproduce some standard findings on sensory modalities. To improve parameter validity, rigorous likelihood-free inverse modeling methods could be used [34]. More importantly, more sophisticated noise models, especially addressing signal-dependent noise, should be considered.

**CONTROL MODELS** To account for known aspects of temporal control better (for instance, for applications requiring synchronicity or target interception), models of sensory synchronization should be considered (e.g., [4, 61]). To extend to aiming tasks, a major concern with input devices, one would have to consider, among other factors, modeling of the whole kinematic chain, not just a single bone, as well as hand–eye coordination, in a closed-loop fashion. This would require explaining how internal models participate in real-time control and how control occurs at multiple levels [21]. Considering closed-loop control is necessary also for advanced button designs, such as double activation buttons, including camera shutter-release buttons and force-sensitive buttons of gamepads. The use of such input devices often involves active haptic exploration [32] – for example, pressing a double-activating button midway to determine how to activate it.

**SENSATION** Our model of sensory processing focuses on the effects of noise on timing estimates. To investigate the effect of materials on button-pressing performance, one could look into more sophisticated models of mechanoreception [74] and to understand other cues than timing, such as force, Bayesian models of sensory integration could be investigated.

**STRATEGIES** We were surprised to learn that button-pressing involves rich strategic variability, including such decisions as how forcefully to press, how long to keep the finger down, or how forcefully to raise it. In the future, we should study how strategies can be expressed in the objective function and, more critically, weights obtained empirically for a given user.

## NEUROMECHANIC: A MODELING WORKBENCH

NEUROMECHANIC is released as a modeling workbench used in MATLAB. Button designs and user tasks can be explored by manipulating parameters as presented above in the *Simulations* section. After optimization, the model outputs such values as means and statistical indices for task-level performance (button activation and perceptual error), cognition (p-centers), and dynamics and kinematics (FD, DV, pulp contraction, and muscle forces). A visual simulation of the dynamics is available also. The model and datasets can be found at <http://userinterfaces.aalto.fi/neuromechanics>. A database of 20 FD curves of commercial button types is available too.

## ACKNOWLEDGMENTS

This work was funded by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation program (grant agreement 637991) and by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2017R1C1B2002101).

## REFERENCES

1. Kenichi Akagi. 1992. A computer keyboard key feel study in performance and preference. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 36. SAGE Publications Sage CA: Los Angeles, CA, 523–527.
2. Gisa Aschersleben. 2002. Temporal control of movements in sensorimotor synchronization. *Brain and cognition* 48, 1 (2002), 66–79.
3. Gisa Aschersleben, Jörg Gehrke, and Wolfgang Prinz. 2004. A psychophysical approach to action timing. *Psychophysics beyond sensation: Laws and invariants of human cognition* (2004), 117–136.
4. Gisa Aschersleben and Wolfgang Prinz. 1995. Synchronizing actions with events: The role of sensory information. *Attention, Perception, & Psychophysics* 57, 3 (1995), 305–317.
5. W Ross Ashby. 1961. *An introduction to cybernetics*. Chapman & Hall Ltd.
6. Myroslav Bachynskyi, Gregorio Palmas, Antti Oulasvirta, and Tino Weinkauff. 2015. Informing the design of novel input methods with muscle coactivation clustering. *ACM Transactions on Computer-Human Interaction (TOCHI)* 21, 6 (2015), 30.
7. Joanna Bergstrom-Lehtovirta, Antti Oulasvirta, and Stephen Brewster. 2011. The effects of walking speed on target acquisition on a touchscreen interface. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. ACM, 143–146.
8. W Thomas Bourbon. 1995. Perceptual control theory. *Comparative approaches to cognitive science* (1995), 151–172.
9. Hans Brunner and Rose Mae Richardson. 1984. Effects of keyboard design and typing skill on user keyboard preferences and throughput performance. In *Proceedings of the Human Factors Society Annual Meeting*, Vol. 28. SAGE Publications Sage CA: Los Angeles, CA, 267–271.
10. Etienne Burdet, David W Franklin, and Theodore E Milner. 2013. *Human robotics: neuromechanics and motor control*. MIT press.
11. Timothy A Carey, Warren Mansell, and Sara J Tai. 2014. A biopsychosocial model based on negative feedback and control. *Frontiers in human neuroscience* 8 (2014).
12. Géry Casiez, Thomas Pietrzak, Damien Marchal, Sébastien Poulmane, Matthieu Falce, and Nicolas Roussel. 2017. Characterizing Latency in Touch and Button-Equipped Interactive Systems. In *UIST 2017*.
13. Andy Clark. 2013. Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behavioral and Brain Sciences* 36, 3 (2013), 181–204.
14. Matthew JC Crump and Gordon D Logan. 2010. Warning: This keyboard will deconstruct—The role of the keyboard in skilled typewriting. *Psychonomic bulletin & review* 17, 3 (2010), 394–399.
15. Hannes Eisler, Anna D Eisler, and Åke Hellström. 2008. Psychophysical issues in the study of time perception. *Psychology of time* (2008), 75–109.
16. Roger M Enoka. 2008. *Neuromechanics of human movement*. Human kinetics.
17. Marc O Ernst. 2006. A Bayesian view on multimodal cue integration. *Human body perception from the inside out* 131 (2006), 105–131.
18. Marc O Ernst and Heinrich H Bülthoff. 2004. Merging the senses into a robust percept. *Trends in cognitive sciences* 8, 4 (2004), 162–169.
19. PSE Farrell, JG Hollands, MM Taylor, and HD Gamble. 1999. Perceptual control and layered protocols in interface design: I. Fundamental concepts. *International Journal of Human-Computer Studies* 50, 6 (1999), 489–520.
20. Jose Gaspar, Mihail Fontul, Elsa Henriques, and Arlindo Silva. 2017. Haptics of in-car radio buttons and its relationship with engineering parameters. *International Journal of Industrial Ergonomics* 59 (2017), 29 – 45.
21. Werner Goebel and Caroline Palmer. 2009. Synchronization of timing and motion among performing musicians. *Music Perception: An Interdisciplinary Journal* 26, 5 (2009), 427–438.
22. Hiroaki Gomi. 2008. Implicit online corrections of reaching movements. *Current opinion in neurobiology* 18, 6 (2008), 558–564.
23. Taku Hachisu and Hiroyuki Kajimoto. 2017. Vibration Feedback Latency Affects Material Perception during Rod Tapping Interactions. *IEEE transactions on haptics* (2017).
24. AV Hill. 1938. The heat of shortening and the dynamic constants of muscle. *Proceedings of the Royal Society of London B: Biological Sciences* 126, 843 (1938), 136–195.
25. Eve Hoggan, Stephen A Brewster, and Jody Johnston. 2008. Investigating the effectiveness of tactile feedback for mobile touchscreens. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 1573–1582.
26. Jakob Hohwy. 2013. *The predictive mind*. Oxford University Press.
27. Michael J Hove and Peter E Keller. 2010. Spatiotemporal relations and movement trajectories in visuomotor synchronization. *Music Perception: An Interdisciplinary Journal* 28, 1 (2010), 15–26.
28. Bruce Hutcheon and Yosef Yarom. 2000. Resonance, oscillation and the intrinsic frequency preferences of neurons. *Trends in neurosciences* 23, 5 (2000), 216–222.



29. Curt B Irwin and Mary E Sesto. 2012. Performance and touch characteristics of disabled and non-disabled participants during a reciprocal tapping task using touch screen technology. *Applied Ergonomics* 43, 6 (Nov. 2012), 1038–1043.
30. Richard J Jagacinski and John M Flach. 2003. *Control theory for humans: Quantitative approaches to modeling performance*. CRC Press.
31. Roland S Johansson and J Randall Flanagan. 2009. Coding and use of tactile signals from the fingertips in object manipulation tasks. *Nature reviews. Neuroscience* 10, 5 (2009), 345.
32. Lynette A Jones and Susan J Lederman. 2006. *Human hand function*. Oxford University Press.
33. Topi Kaaresoja, Stephen Brewster, and Vuokko Lantz. 2014. Towards the temporally perfect virtual button: touch-feedback simultaneity and perceived quality in mobile touchscreen press interactions. *ACM Transactions on Applied Perception (TAP)* 11, 2 (2014), 9.
34. Antti Kangasrääsiö, Kumaripaba Athukorala, Andrew Howes, Jukka Corander, Samuel Kaski, and Antti Oulasvirta. 2017. Inferring Cognitive Models from Data using Approximate Bayesian Computation. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 1295–1306.
35. Mitsuo Kawato. 1999. Internal models for motor control and trajectory planning. *Current opinion in neurobiology* 9, 6 (1999), 718–727.
36. Jeong Ho Kim, Lovenoor Aulck, Michael C Bartha, Christy A Harper, and Peter W Johnson. 2014. Differences in typing forces, muscle activity, comfort, and typing performance among virtual, notebook, and desktop keyboards. *Applied ergonomics* 45, 6 (2014), 1406–1413.
37. Sunjun Kim, Jeongmin Son, Geehyuk Lee, Hwan Kim, and Woohun Lee. 2013. TapBoard: making a touch screen keyboard more touchable. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 553–562.
38. Sung Soo Kim, Arun P Sripati, and Sliman J Bensmaia. 2010. Predicting the timing of spikes evoked by tactile stimulation of the hand. *Journal of neurophysiology* 104, 3 (2010), 1484–1496.
39. David C Knill and Alexandre Pouget. 2004. The Bayesian brain: the role of uncertainty in neural coding and computation. *TRENDS in Neurosciences* 27, 12 (2004), 712–719.
40. Konrad P Körding and Daniel M Wolpert. 2006. Bayesian decision theory in sensorimotor control. *Trends in cognitive sciences* 10, 7 (2006), 319–326.
41. Byungjoo Lee, Qiao Deng, Eve Hoggan, and Antti Oulasvirta. 2017. Boxer: A Multimodal Collision Technique for Virtual Objects. In *Proceedings of 19th ACM International Conference on Multimodal Interaction (ICMI'17)*. ACM.
42. Byungjoo Lee and Antti Oulasvirta. 2016. Modelling error rates in temporal pointing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1857–1868.
43. Byungjoo Lee, Kim Sunjun, Antti Oulasvirta, Jong-In Lee, and Eunji Park. 2018. Moving Target Selection: A Cue Integration Model. In *Proceedings of the 36th Annual ACM Conference on Human Factors in Computing Systems (CHI'18)*. ACM Press.
44. Sung-Hee Lee, Eftychios Sifakis, and Demetri Terzopoulos. 2009. Comprehensive biomechanical modeling and simulation of the upper body. *ACM Transactions on Graphics (TOG)* 28, 4 (2009), 99.
45. Sung-Hee Lee and Demetri Terzopoulos. 2006. Heads up!: biomechanical modeling and neuromuscular control of the neck. In *ACM Transactions on Graphics (TOG)*, Vol. 25. ACM, 1188–1198.
46. James R. Lewis, Kathleen M. Potosnak, and Regis L. Magyar. 1997. Chapter 54 - Keys and Keyboards. In *Handbook of Human-Computer Interaction (Second Edition)* (second edition ed.), Marting G. Helander, Thomas K. Landauer, and Prasad V. Prabh (Eds.). North-Holland, Amsterdam, 1285 – 1315.
47. Jon Moeller and Andruid Kerne. 2012. ZeroTouch: an optical multi-touch and free-air interaction architecture. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2165–2174.
48. Jon Moeller, Nic Lupfer, Bill Hamilton, Haiqiao Lin, and Andruid Kerne. 2011. intangibleCanvas: free-air finger painting on a projected canvas. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*. ACM, 1615–1620.
49. John Morton, Steve Marcus, and Clive Frankish. 1976. Perceptual centers (P-centers). *Psychological Review* 83, 5 (1976), 405.
50. Jörg Müller, Antti Oulasvirta, and Roderick Murray-Smith. 2017. Control Theoretic Models of Pointing. *ACM Transactions on Computer-Human Interaction (TOCHI)* 24, 4 (2017), 27.
51. Mark Nagurka and Richard Marklin. 2005. Measurement of stiffness and damping characteristics of computer keyboard keys. *Journal of dynamic systems, measurement, and control* 127, 2 (2005), 283–288.
52. Kiisa Nishikawa, Andrew A Biewener, Peter Aerts, Anna N Ahn, Hillel J Chiel, Monica A Daley, Thomas L Daniel, Robert J Full, Melina E Hale, Tyson L Hedrick, and others. 2007. Neuromechanics: an integrative approach for understanding motor control. *Integrative and Comparative Biology* 47, 1 (2007), 16–54.

53. Dan Odell and Eric Faggin. 2014. The typing performance and preference costs of reducing tactile feedback and tactile landmarks in tablet keyboards. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 58. SAGE Publications Sage CA: Los Angeles, CA, 1790–1794.
54. Antti Oulasvirta and Joanna Bergstrom-Lehtovirta. 2010. A simple index for multimodal flexibility. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1475–1484.
55. Denham L Phipps and Don Harris. 2014. Perceptual control and feedback control in the analysis of complex tasks. *Theoretical Issues in Ergonomics Science* 15, 5 (2014), 505–516.
56. William T Powers. 2009. *Perceptual Control Theory: Science & Applications: A Book of Readings*. Living Control Systems Publ.
57. William T Powers and William T Powers. 1973. *Behavior: The control of perception*. Aldine Chicago.
58. Robert G Radwin and One-Jang Jeng. 1997. Activation force and travel effects on overexertion in repetitive key tapping. *Human Factors* 39, 1 (1997), 130–140.
59. Robert G Radwin and Barry A Ruffalo. 1999. Computer key switch force-displacement characteristics and short-term effects on localized fatigue. *Ergonomics* 42, 1 (1999), 160–170.
60. David Rempel, Elaine Serina, Edward Klinenberg, Bernard J Martin, Thomas J Armstrong, James A Foulke, and Sivakumaran Natarajan. 1997. The effect of keyboard keyswitch make force on applied force and finger flexor muscle activity. *Ergonomics* 40, 8 (1997), 800–808.
61. Bruno H Repp. 2005. Sensorimotor synchronization: a review of the tapping literature. *Psychonomic bulletin & review* 12, 6 (2005), 969–992.
62. Stefan Schaal and Nicolas Schweighofer. 2005. Computational motor control in humans and robots. *Current opinion in neurobiology* 15, 6 (2005), 675–682.
63. Steven C Seow. 2005. Information theoretic models of HCI: a comparison of the Hick-Hyman law and Fitts' law. *Human-Computer Interaction* 20, 3 (2005), 315–352.
64. Anil K Seth. 2014. The cybernetic Bayesian brain. In *Open Mind*. Open MIND. Frankfurt am Main: MIND Group.
65. Tobias Siebert, Christian Rode, Walter Herzog, Olaf Till, and Reinhard Blickhan. 2008. Nonlinearities make a difference: comparison of two common Hill-type models with real muscle. *Biological cybernetics* 98, 2 (2008), 133–143.
66. Srinath Sridhar, Anna Maria Feit, Christian Theobalt, and Antti Oulasvirta. 2015. Investigating the dexterity of multi-finger input for mid-air text entry. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 3643–3652.
67. Kim Sunjun, Byungjoo Lee, and Antti Oulasvirta. 2018. Impact Activation Improves Rapid Button Pressing. In *Proceedings of the 36th Annual ACM Conference on Human Factors in Computing Systems (CHI'18)*. ACM Press.
68. Lena H Ting and J Lucas McKay. 2007. Neuromechanics of muscle synergies for posture and movement. *Current opinion in neurobiology* 17, 6 (2007), 622–628.
69. Kjerstin Torre and Ramesh Balasubramaniam. 2009. Two different processes for sensorimotor synchronization in continuous and discontinuous rhythmic movements. *Experimental Brain Research* 199, 2 (2009), 157–166.
70. Michel Treisman. 1963. Temporal discrimination and the indifference interval: Implications for a model of the "internal clock". *Psychological Monographs: General and Applied* 77, 13 (1963), 1.
71. Macy Ann Valk. 1985. An experiment to study touchscreen button design. In *Proceedings of the Human Factors Society Annual Meeting*, Vol. 29. SAGE Publications Sage CA: Los Angeles, CA, 127–131.
72. Michael Wiertlewski and Vincent Hayward. 2012. Mechanical behavior of the fingertip in the range of frequencies and displacements relevant to touch. *Journal of biomechanics* 45, 11 (2012), 1869–1874.
73. John Z Wu, Ren G Dong, S Rakheja, and AW Schopper. 2002. Simulation of mechanical responses of fingertip to dynamic loading. *Medical Engineering & Physics* 24, 4 (2002), 253–264.
74. John Z Wu, Ren G Dong, S Rakheja, AW Schopper, and WP Smutz. 2004. A structural fingertip model for simulating of the biomechanics of tactile sensation. *Medical engineering & physics* 26, 2 (2004), 165–175.