

Improving Scrolling Devices with Document-Length-Dependent Gain

Andy Cockburn[†], Philip Quinn[†], Carl Gutwin[‡], Stephen Fitchett[†]

[†]Computer Science, University of Canterbury, Christchurch, New Zealand

andy@cosc.canterbury.ac.nz, philip.quinn@canterbury.ac.nz, stephen.fitchett@pg.canterbury.ac.nz

[‡]Computer Science, University of Saskatchewan, Saskatoon, Canada, gutwin@cs.usask.ca

ABSTRACT

We describe a method for applying gain to events reported by scrolling input devices such as scroll wheels. By treating document length as an input to our gain functions, the method allows rapid document traversal regardless of document length; it also allows slow and precise scroll control at shorter distances. An initial experiment characterises four diverse scrolling input devices – a standard ‘notched’ scrollwheel, a high performance ‘inertial’ wheel, an isometric scrolling joystick, and a trackpad – and the results are used to calibrate several gain function parameters. A second experiment validates the method, showing that it allows faster scrolling in long and short documents than current scrolling-device gain methods, and that subjective preferences favour it.

Author Keywords

Scrolling; control-display gain; empirical studies.

ACM Classification Keywords

H.5.2 [User Interfaces]: *Input devices and strategies*;

INTRODUCTION

Scrolling provides a simple means for working with documents that are too large to be conveniently displayed within a single view. Commensurate with its importance, scrolling interactions have been extensively researched, producing innovative devices, techniques, and behaviours. Surprisingly, though, there has been little public research into one of the most basic and universal scrolling problems – that of supporting appropriate gain functions to allow both precise repositioning actions as well as rapid movement across large distances. Although most pointing devices include scroll controls (e.g., scrollwheels) there has been little research into optimising their performance.

Scrolling and mouse-based targeting are similar in that both have a wide range of control requirements – from small precise pixel actions to long range movement. Yet gain functions for scrolling are relatively unstudied (with Hinckley [11] being a notable exception, reviewed later).

Permission to make digital or hard copies of all or part 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. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2012, May 5–10, 2012, Austin, Texas, USA.

Copyright 2012 ACM 978-1-4503-1015-4/12/05...\$10.00.

For pointing, however, gain studies are common, and they are known to improve performance [6, 7].

Two main problems complicate the design of effective scroll gain functions. First, many scrolling input devices allow only coarse levels of control. For example, the range of physical movement enabled by most mouse scrollwheels is approximately 10mm linearly without ‘clutching’ (the process of temporarily disengaging from the wheel in order to reposition the finger). Furthermore, this limited movement is often discretised into 5 or 6 coarse ‘notches’. By contrast, clutch-free mouse pointing extends over approximately ten times this linear distance, on two dimensions, with hundreds of registered points per cm. The second problem complicating the design of scroll gain functions is that the linear range of control requirements for scrolling is much larger than that of cursor pointing. Pointing requires at most a few thousand pixels be addressable on the x or y dimensions. Documents, however, can be thousands of pages long, with each page containing roughly a thousand linear pixels, any of which might be considered a target for positioning the scrolled display.

The conflict between the low fidelity of scrolling input controls and the wide range of possible scrolling targets means that clutching with scrolling input devices is often pronounced, causing users to frenetically repeat scrolling actions, exhibited as ‘clawing’ a scrollwheel or ‘scratching’ a touchscreen. Consequently, many scroll devices, such as standard notched scrollwheels, are best used only for short range movement, and when longer movements are required users should switch to an alternative control, such as dragging the scroll thumb (as part of a scrollbar). However, expecting users to make a distance-based choice between scrolling methods is inefficient: decisions take time, they can be made incorrectly, and they require that the user switch their attention from the task to the interface. As a result, users may avoid the decision and try to use a single technique regardless of its comparative inefficiency for a particular range of distances (e.g., people may often use the scroll wheel for long distances, even when this takes far longer than using the thumb). Given that devices such as scrollwheels are already in common use, it is reasonable to expect that these controls should be usable for all scrolling needs – no one would consider using two mice for pointing (one for near targets and another for far ones), yet curiously, this is the equivalent expectation for scrolling.

Two observations – that documents vary greatly in length, and that scrolling mechanisms are inefficient at long distances – lead us to the idea of making gain functions that are dependent on document length. Document-length-dependent (DLD) gain is a simple method that can substantially improve the performance of scrolling input devices. Our method of evaluation involves a rapid calibration process to understand device characteristics, which could be performed by the device manufacturer or by individuals seeking tailored performance. The gain function allows pixel-precise movement as well as rapid document traversal, regardless of length. Experiments validate the technique’s effectiveness across a diverse range of devices.

The specific contributions of this paper are as follows:

- An empirical characterization of human performance with different scrolling devices: a notched scrollwheel, an isometric ‘ScrollPoint’ joystick, an inertial scrollwheel, and two-finger trackpad scrolling.
- Details of our document-length-dependent gain function.
- Empirical results demonstrating that performance and preferences are improved by our gain function.

SCROLLING FRAMEWORK & RELATED WORK

Abowd and Beale’s interaction framework [1] subdivides interaction into translations from the user’s motor actions to input events, from input events to system state, from state to output, and from output to perception. We use this framework to clarify scrolling activities and to classify prior research.

User → Input: Motor action

Several previous studies have examined input device capabilities, including Buxton’s analysis of position, motion, and pressure input for controlling multiple input dimensions [4], Card, Mackinlay and Robertson’s morphological analysis of input [5], and Jacob *et al.*’s mapping between input device and user task [15]. These analyses focus on pointing requirements, and none share our specific focus on document scrolling.

Most scrolling devices are isotonic, using either linear or rotary movement to control absolute position, relative position, or rate of movement. Absolute position control assigns specific meaning to fixed locations: for example, touching the bottom of a trackpad slider might map directly to the bottom of a document. Relative position control normally involves manipulating instant-to-instant displacement (e.g., pointing with a mouse), but it can also be used to control displacement between a ‘marked’ location and a tracked one, as is the case when using middle-button autoscroll to control scroll speed (e.g., [9]). With *rate of movement* control the velocity of interaction is the primary input parameter, with examples including the speed of gestural flicks on a touchscreen (e.g., [2]) and the speed of rotary movement (e.g., [16, 17]). While most devices are isotonic, Zhai’s IBM ‘ScrollPoint’ mouse uses

an isometric knob for high performance scrolling [11, 19], making scroll speed proportional to the force applied to the knob. One benefit of rotary and isometric input methods is that they can be continually operated without clutching.

Input → System: Device/driver events

Input devices translate motor actions into electronic events which are interpreted and modified by low level driver software before being forwarded to applications. Quantization and driver gain critically affect this process.

Quantization problems occur when input discretisation impedes performance. This arises in many aspects of scrolling, such as mapping from a constrained range of scrollbar positions to a large range of potential views, or the coarse scroll increments supported by notched scroll wheels. The impact of quantization has been examined in several pointing studies (e.g., [7]), but we are unaware of similar analysis for scrolling.

Scrolling devices report raw events corresponding to the degree of a user’s manipulation, which is translated into a magnitude of movement m (measured in number of lines or pixels) by software device drivers. The drivers can also attend to the reported m value and the time since the last event (Δt) and use these to apply transformations to the device events, typically using gain functions to improve performance. While there have been many studies of pointing gain (e.g., [6, 7]), there has been little public research on scroll gain, with Hinckley *et al.* [11] being a notable exception.

Hinckley *et al.* [11] describe a scrollwheel gain function (Equation 1) and a methodology for evaluating scrolling performance. Their experiments demonstrate that their gain function improves scrollwheel performance.

$$\Delta y = k_1(1 + k_2\Delta t)^\alpha \quad \text{Equation 1.}$$

In this equation, Δy is the resultant gain, k_1 , k_2 , and α are constants, and Δt is the time between scroll events. The Δy gain value is used to multiply event magnitude m . For negative values of α , this function caps gain to a maximum value of k_1 . Recognising the limitation of capped gain for long distance scrolling, Hinckley *et al.* [12] patented ‘distance-based accelerated scrolling’, in which rapid repeated scrollwheel clutches have an additive effect on gain, while pauses or direction reversals cancel it. This requires calibration of timeout values and threshold levels to distinguish small pauses from clutched reacquisition of the scrollwheel. We are unaware of any evaluation of the method, and we include it in Experiment 2.

System→Output→User: Display update and perception

Applications are responsible for updating the display in response to events reported from the driver. The default behaviour is to adjust the viewport by the magnitude requested, but applications can apply other effects based on particular needs. For example, ‘speed dependent automatic

zooming’ [13] reduces motion blur by automatically zooming away from documents at high scroll speeds, and ‘content aware scrolling’ [14] allows documents to stipulate particular scroll paths based on document content. Techniques such as these change the basic scrolling behaviour, adding side effects to scroll-based movement, such as zooming or path alteration. In contrast, our goal with DLD-gain is to improve performance of basic scrolling using ubiquitous devices such as scroll wheels.

DOCUMENT-LENGTH-DEPENDENT GAIN

In Human-Computer Interaction, gain functions are used to amplify the raw signals from a device to assist the user in achieving their tasks. The level of gain calculated by the function(s) depends on some input parameter. For example, ‘pointer acceleration’ [7] is a gain function whereby pointer movement is amplified dependent on mouse velocity – when the mouse moves quickly, the user is assumed to want to move far, so higher gain assists them in doing so.

The specific goals of our document-length-dependent scrolling gain functions are as follows:

Pixel-level control. Recent scrolling devices and drivers allow smooth document movement by reporting high resolution (sub-line) scroll increments. Our gain functions enable smooth sub-line scroll increments even when the minimal scroll increment reported by the device/driver is an integral number of lines – the functions smooth out rough devices.

Constant time for full document traversal. Regardless of document length, the gain function should allow the entire document to be traversed (from start to end) in a near-constant time. Without this property device efficiency is likely to decrease as document length increases.

Efficient regardless of range. The gain function must remain efficient across all movement ranges – lines, paragraphs, pages, chapters, and entire books.

Subjectively appealing. Users must find the gain-modified device satisfying and comfortable for all movement ranges.

Details of the gain method

Our gain method uses three functions to map from device velocity to a resultant gain level: a slow movement function (SMF), a fast proportional function (FPF), and a slow-to-fast transition function (SFTF), described below.

As mentioned in related work, two parameters can be derived from each scrolling event – the magnitude of the movement (m) and the time since the last event (Δt). From these values we calculate a device velocity for each event ($v=m/\Delta t$), which is input to our functions to determine a gain level (g). The gain level is then used as a multiplier to the initial event magnitude to calculate a resultant scrolling distance ($d=g \times m$). This approach differs to Hinckley *et al.* [11] in that our functions calculate gain based on device velocity rather than Δt alone. Doing so allows the functions

to receive a wider range of input values, particularly at high scroll-speeds (demonstrated by the calibration experiment). The three component functions are described below.

Slow movement function. SMF controls gain (g_s) during precise scrolling actions – it maps from the device velocity (v) to gain level using Equation 2. Figure 1 shows that this gives a rapid increase through fractional gain levels up to a capped constant level k_s (set to 1 by default). Fractional gain allows devices that report line-based scroll increments to support pixel-level smooth scrolling. The parameter α determines the rate of gain increase, which could be set through a user preferences dialog. Note that Equation 2 is independent of document length, meaning that slow document movement is consistently controlled across all documents.

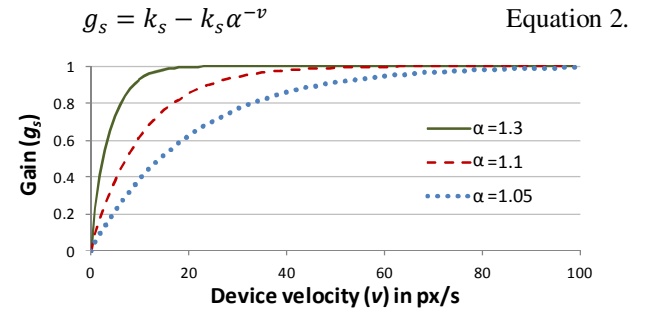


Figure 1. The slow movement function allows line-based devices to offer pixel scroll control through fractional gain.

Fast proportional function. The FPF function calculates gain (g_f) based on the *proportion* of the document that is visible in the viewport (see Equation 3). High gain levels are therefore possible for long documents or for documents that are viewed when closely zoomed. This is analogous to the behaviour of the scroll thumb – the number of document lines traversed when dragging the thumb depends on the ratio of document length to viewport size. The constant k_f allows adjustment of the maximum scrolling gain, and can be used to account for differences in device report granularity and for user preferences.

$$g_f = k_f \times \frac{\text{document length}}{\text{viewport size}} \quad \text{Equation 3.}$$

$$p_s = 1 - p_f \quad \text{Equation 4.}$$

$$p_f = \begin{cases} 0, & \text{if } v/v_{max} < \text{initial} \\ \frac{v/v_{max} - \text{initial}}{\text{final} - \text{initial}}, & \text{if } \text{initial} \leq \frac{v}{v_{max}} \leq \text{final} \\ 1, & \text{if } v/v_{max} > \text{final} \end{cases} \quad \text{Equation 5.}$$

$$g = (p_s \times g_s) + (p_f \times g_f) \quad \text{Equation 6.}$$

Slow-to-fast transition. The SFTF function controls the transition between slow-movement and fast-proportional functions. It does so by linearly transitioning between the proportional contribution of the SMF and FPF gain functions (p_s , p_f), based on the proportion of maximum device velocity (v/v_{max}). This transition is shown in Figure 2, Equation 4 and the piecewise Equation 5. The

transition to using FMF begins when the device velocity exceeds an *initial* threshold, and it is complete at a *final* threshold. The gain calculation is shown in Equation 6.

The overall effect of these functions is similar to an exponential mapping from v to g , but the gain onset is postponed until high velocities to assist users with document control when moving short distances.

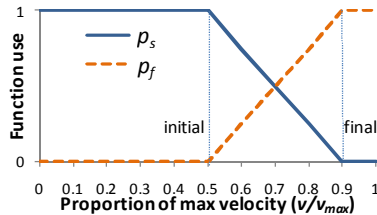


Figure 2. Slow-to-fast transition function, which controls the proportional contribution of slow and fast functions.

The potential drawback of the technique is that the fast proportional function (Equation 3) will produce high gain levels in long documents, which may cause control problems (exhibited as overshooting) when the *initial* threshold is accidentally attained. This effect will be more severe in long documents, since the interpolation gradient between the slow and fast functions will be steeper.

To investigate and evaluate the idea of document-length-dependent gain, we carried out two experiments. The objective of Experiment 1 was to understand important aspects of human control with four substantially different scrolling input devices, to characterise their event streams, and to use this data to calibrate the DLD-gain parameters identified in the previous section. Experiment 2 then validated DLD-gain by analysing human performance when using it; this study also examines the trade-off between fast long-distance movement and a steep interpolation function.

EXPERIMENT 1: CALIBRATING SCROLLING DEVICES

Calibrating DLD-gain would be straightforward if all users had exactly the same physical capabilities, if these capabilities were identically expressed in upward and downward scroll directions, if device event streams were smooth, and if all devices produced exactly the same range of values in their event reports. None of these are likely to be true. Experiment 1 therefore answers several questions that are used to calibrate DLD-gain parameters, as follows:

Q1: Is it necessary to separately calibrate v_{max} for different participants, directions, and devices? Equation 5 shows that v_{max} is a critical value controlling the transition between slow precise movement and fast proportional movement. If there is little variation between participants, directions, and devices then a single value would be sufficient for all, but if variation is substantial then individual calibration is necessary for participants, directions and devices.

Q2: Is smoothing necessary to reduce noise in event streams? If the m and Δt values reported by a device driver

are substantially unstable while the user attempts to control the device at a constant velocity, then the resultant scrolling behaviour will be jerky, particularly at high gain levels. By understanding the characteristics of the event stream, simple procedures may adequately smooth event reports.

Q3: Is it necessary to diminish events for over-sensitive devices? The slow movement function (Equation 2) is capable of generating fractional gain levels (<1). These low gain levels allow pixel scrolling with devices that report integral line scroll increments, but importantly, they should also improve control of ‘twitchy’ devices that report unexpectedly fast scroll speeds for small motor effects.

The experiment answers these questions using four scrolling devices that use very different control mechanisms: a low-end traditional ‘notched’ scrollwheel (Microsoft Wheel Mouse Optical), an ‘inertia’ free-wheeling scrollwheel (Logitech M905), two-finger scrolling on a *trackpad* (Apple Magic Trackpad), and a mouse-mounted *isometric* scroll knob (IBM ScrollPoint).

Participants and apparatus

Thirteen university students (3 female; mean age 23.5, s.d. 1.9) took part in the experiment, all of whom used computers for at least 7 hours per week. Their participation lasted approximately 40 minutes.

The experiment ran on an Apple iMac with a 21.5” display at 1920×1080 native resolution. The software was written in JavaScript and ran in Firefox 5. The drivers for each device were configured to use their lowest gain settings; we confirmed that these settings produced constant gain (no acceleration) by tracing function calls in the driver code¹.

Procedure

All participants used the four scrolling devices in both scrolling directions (up and down), completing all trials with one device before proceeding to the next (order counter-balanced). Participants completed NASA-TLX [10] worksheets after each device. With each device the procedure consisted of two repetitions (one for scrolling downward, one for upward) of the following activities: *familiarisation*, *v_{max} calibration*, and *slow scrolling*.

Familiarisation with each device/direction combination consisted of four repetitions of operating the device to match the left-to-right velocity of a series of orange vertical lines shown inside a progress bar. Each vertical line traversed the progress bar in three seconds. The participant’s control of scroll velocity was indicated by progressively shading the progress bar grey, with a green leading edge. An abstract representation of scroll velocity was used (rather than scrolling text) to reduce transfer effects from prior scrolling experience.

¹ Available at: <http://opensource.apple.com/source/IOHIDFamily/IOHIDFamily-315.7.16/IOHIDSystem/IOHIPointing.cpp>

V_{max} calibration. To calibrate maximum velocity with the device/direction combination, participants were prompted to scroll ‘As fast as comfortably possible’. They clicked a ‘Start’ button then began operating the device as fast as possible. A progress bar automatically filled at a rate of ¼ per second. This process was repeated four times in each direction, with the first trial discarded as familiarisation.

Slow scrolling. The final condition prompted users to scroll “smoothly and slowly to match the shown rate”, with a document displaying pixel-scrolling at 2.5 lines per second (40 px/s). They repeated the four second task four times.

Results

Q1: Is separate *v_{max}* calibration needed?

We analyse *v_{max}* variation between directions and devices using a 4x2 repeated-measures analysis of variance for factors *input device* and *direction*. The dependent variable is the average speed during each trial (px/s). We applied the Aligned Rank Transform for nonparametric factorial analysis [18] to accommodate a non-normal distribution.

The isometric device reported dramatically higher maximum scroll speeds (mean 12358 px/s) than the other three devices at 1769, 568 and 431 px/s with *inertia*, *notch*, and *trackpad* respectively ($F_{3,36} = 48.3, p < .001$). *Direction* showed a significant main effect, with mean *down* speed (4045 px/s) faster than *up* (3518 px/s): $F_{1,12} = 52.2, p < .001$. A *device* x *direction* interaction ($F_{3,36} = 8.85, p < .001$), was caused by *inertia* having a larger difference between *up* and *down* directions than other devices.

These results demonstrate that device- and direction-dependent *v_{max}* calibration is necessary.

There was also substantial cross-participant variation in *v_{max}*, particularly with *inertia* and *isometric* devices. Figure 3 shows the range, median, mean, and interquartile range of *v_{max}* values across participants with the devices, suggesting that individual *v_{max}* calibration is important.

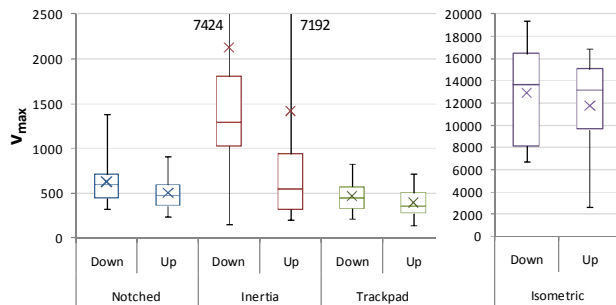


Figure 3. Box-whisker plots characterising cross participant variation in *v_{max}* values with the four devices.

Q2: Is smoothing necessary?

Figure 4 shows typical event streams with each of the four devices during *v_{max}* trials. It illustrates that clutching is prevalent with *notched* and *trackpad* (the velocity frequently decreases to zero while the fingers are

repositioned on the device), and that users were better able to maintain a constant velocity with *inertia* and *isometric*.

The *trackpad* and *isometric* event streams frequently included paired ‘corrective’ events of small magnitude with extremely small Δt values (typically ~1ms), resulting in extremely high peak velocities. We account for these by summing Δt and *m* across events within a 10msec window.

Two further steps were taken to smooth DLD-gain’s performance in response to noisy event streams. First, our implementation sets *v_{max}* to the 70th percentile of velocity attained during maximum velocity calibration trials. Doing so means that *v_{max}* is an attainable velocity, rather than one that can only be accidentally hit when the device reports a ‘spike’. Second, to reduce the tendency for high variance in event stream velocities to produce jerky post-gain velocities, we average DLD-gain (*g*) over a window of *w* event reports (*w*=7 in our implementation). The event window is reset if no events are received for *w_t* seconds (0.5s) and when the scroll direction reverses.

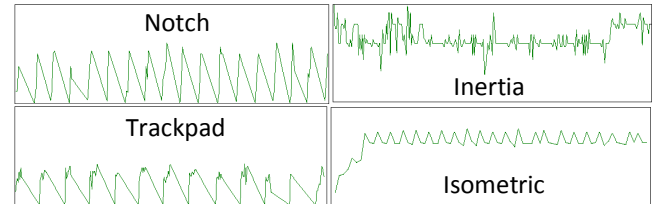


Figure 4. Indicative event streams from the four devices (magnitude *m* on y-axis, time on x-axis).

Q3: Is it necessary to diminish events for sensitive devices?

Slow trials involved operating the devices to match the velocity of a document shown scrolling at 40 px/s. Event stream analysis showed that participants operated three of the devices at roughly twice the intended speed: 82, 69 and 82 px/s with *notch*, *inertia*, and *trackpad*. *Isometric* was extremely fast at 4835 px/s, which suggests that users will have difficulty with slow and precise movement unless its event reports are diminished – confirmed by Experiment 2.

Subjective responses

NASA-TLX worksheet responses [10] on a five point scale from low (1) to high (5) showed significant differences (Friedman χ^2 test, $p < .05$) between devices for physical effort (means of 3.8, 2.8, 3.4, and 2.1 with *notch*, *inertia*, *trackpad* and *isometric* respectively), temporal demand (means of 3.3, 3.4, 3.1, and 2.2), and hard work (3.5, 3.5, 3.5, 2.1). The absence of clutching with *isometric* best explains its low perceived workload. However, participant comments also stated that *isometric* was “too sensitive to control, need to highly concentrate on it”, and “I never feel like I have good control even if I can match a target rate”.

Resultant calibration

Table 1 shows the DLD-gain parameters used in Experiment 2 for the three devices. These values were attained by first finding *k_s* and *a* values that allowed precise control for line and paragraph movement, then finding *k_f*,

initial and *final* values that allowed rapid long range movement. Experiment 2 compares DLD-gain performance with ‘*hinckley*’ gain (the best scroll gain function we are aware of). We used Hinckley *et al.*’s function [11], supplemented with their “additive” technique [12], which accumulates gain across rapidly repeated clutching actions. ‘*Hinckley*’ parameters for Equation 1 are also shown in Table 1, established through a similar process.

	<i>Hinckley</i>			<i>DLD</i>				
	k_1	k_2	α	k_s	α	k_f	<i>initial</i>	<i>final</i>
<i>notched</i>	4	8	-2.5	5	1.005	1	0.9	1.0
<i>inertia</i>	8	16	-3	2	1.005	0.2	0.5	1.0
<i>isometric</i>	10	15	-3	0.1	1.003	0.016	0.8	0.9

Table 1. Gain parameters for *hinckley* and *DLD* conditions.

EXPERIMENT 2: VALIDATING DLD-GAIN

We validate DLD-gain in a three-part experiment. The first part compares performance with three devices using DLD-gain against that of the same devices using Hinckley’s gain functions. It uses an extremely long document (20,000 lines; 333 pages at 60 lines per page) that will cause DLD-gain to produce very high gain levels when the user exceeds the *initial* speed threshold – DLD-gain is therefore likely to be ‘twitchy’ at medium speeds. Part 1’s hypothesis is:

H1: DLD-gain improves performance compared to Hinckley’s gain functions in an extremely long document.

Hinckley’s gain functions are used as the comparator because they represent the state-of-the-art – they are known to improve performance with notched scrollwheels [11], and our pilot testing demonstrated that they radically improved performance with inertia and isometric devices, particularly for long distances.

Part 2 of the experiment analyses user performance with DLD-gain when scrolling in shorter documents, testing the following hypothesis.

H2: scrolling performance with DLD-gain improves as document length decreases.

Finally, when switching between documents of different lengths user performance and preferences may be adversely affected by ‘unpredictable’ gain. This is tested in Part 3:

H3: user performance with DLD-gain is not adversely affected when switching between document lengths.

These hypotheses are tested using Hinckley *et al.*’s method for analysing mechanical aspects of scrolling performance [11], which involves bidirectional tapping between target lines at controlled scroll distances. The target line is indicated by showing a mark at its location in the scroll trough and by highlighting the target line green (Figure 5). A target is acquired by pressing the space bar when the target line rests within a marked central region of the viewport (six lines wide). Pressing the space bar when the target is outside the target region constitutes an error, and

the trial continues until successfully acquired. Completing one trial automatically begins the next, alternating between downward and upward scrolling. We used a wide range of distances representing movement across lines, paragraphs, pages, chapters, and books.

Part 1: Hinckley gain versus DLD-gain

Experiment 2 Part 1 compares performance with three devices when using DLD-gain and when using Hinckley’s gain functions. The devices were *notch*, *inertia*, and *isometric*, with *trackpad* discarded for expediency due to its calibration similarity to *notch* in Experiment 1. As our research objective is to improve performance of scrolling devices, other methods for scrolling (such as the dragging the scroll-thumb) were disabled.

The experiment analyses performance across scrolling distance and direction. Pilot testing demonstrated that long distance scrolling with Hinckley’s function can be prohibitively time-consuming, so to avoid prolonged and frenetic use of the device, trials were capped at 15 seconds.

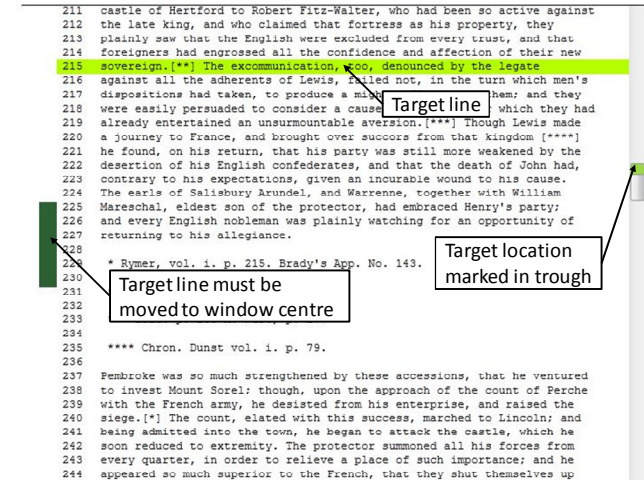


Figure 5. Experimental interface. Tasks involve moving the green target line into the central window region (left).

Design and Procedure

The primary dependent measure is acquisition time, with a secondary dependent measure for number of target overshoots and undershoots (necessitating a change in scrolling direction).

Data are analysed using a 2x3x5x2 ANOVA for within subjects factors *input device*, *gain*, *distance*, and *direction*:

$$\begin{aligned}
 \textit{Gain} &\in \{\textit{hinckley}, \textit{DLD}\} \times \\
 \textit{Input device} &\in \{\textit{notch}, \textit{inertia}, \textit{isometric}\} \times \\
 \textit{Distance} &\in \{7, 49, 343, 2401, 16807\} \times \\
 \textit{Direction} &\in \{\textit{up}, \textit{down}\} \times 3 \textit{ repetitions}
 \end{aligned}$$

With each device, participants completed nine device familiarisation trials with no gain. They then completed a v_{max} calibration for the *up* and *down* directions using the same procedure as Experiment One. The main experimental trials consisted of six selections at each of five levels of

distance, with shorter distances preceding longer ones. The six trials consisted of three bi-directional repetitions (downward then upward thrice), with the first discarded as target familiarisation.

All trials with each *input device* were completed before moving on to the next *input device*, within which, all trials with each *gain* level were completed before moving on to the next gain. Six orders of *input device* and *gain* were balanced using a Latin square.

Participants and apparatus

Sixteen university students (6 female; mean age 25.5, sd. 4.5) took part in the experiment, with participation lasting approximately 40 minutes.

The experiment ran on a Microsoft Window 7 PC with a 22" display at 1680x1050 native resolution. The experimental software was written in JavaScript and ran in Firefox 6. Device drivers were configured to produce constant gain (no scroll acceleration). The document used for the study was 20,000 lines of raw text from David Hume's *A History of England*.

Part 1: Results

At the longest scrolling distance (16807 lines) most trials were automatically terminated before the 15 second time cap with *hinckley* (90%, 62%, and 100% with *notch*, *inertia*, and *isometric* respectively), but comparatively few were automatically terminated with *DLD* (6%, 17%, and 9% respectively). In the following analysis the time cap value of 15 seconds is inserted for all time capped trials (consequently, substantially underestimating times with *hinckley* for distance 16807). *Direction* was found to have no significant effects or interactions, so we collapsed the analysis across *direction* and report results of a 2x3x5 ANOVA for *gain*, *device*, and *distance*.

Mean trial times with *DLD* (5.82s) were significantly faster than *Hinckley* (6.8s), leading us to accept **H1**: $F_{1,15} = 86.5, p < .001$. There was also significant main effect of *device* ($F_{2,30} = 15.8, p < .001$), with *inertia* fastest (6.0s), followed by *notch* (6.1) and *isometric* (6.8). As expected, *distance* showed a significant main effect: $F_{4,60} = 815.6, p < .001$.

Importantly, all interactions were also significant. Figure 6(a-c) separately show the *gain* x *distance* interactions for

each device ($F_{4,60} = 117.0, p < .001$). The figures show that *hinckley* gain was slightly faster than *DLD*-gain for most short scrolling distances. For longer distances, however, *DLD*-gain was much faster, especially when considering that nearly all long distance *hinckley* trials were prematurely terminated by the 15 second cap.

To produce an estimate of how long it would have taken participants to complete the 16807 distance trials using *hinckley*, we used the distance completed during the 15 seconds of scrolling to calculate an average speed, and extrapolated for the remaining distance. This produces a mean estimate of 33 seconds for *notch*, 21 seconds with *inertia*, and 85 second with *isometric*, suggesting that long distance scrolling is impractical using *hinckley* gain. Regardless of *DLD*'s faster overall performance, its slightly worse performance at the 343 line distance is notable (*index of difficulty*=5.6 in Figure 6). We attribute this to participants accidentally hitting the *initial* threshold and consequently experiencing high gain levels due to the extreme document length strongly influencing Equation 3. This explanation is supported by scrolling path analysis, which shows significantly higher target overshooting with *DLD* (passing over the target and changing direction to return). The mean number of overshoots per trial was 0.67 with *hinckley* and 1.14 with *DLD* ($F_{1,15} = 41.0, p < .001$). Note that the number of overshoots characterises interaction but does not necessarily imply a performance problem – for example, a rapid ballistic phase of movement will promote overshooting, while gradual slow target approach will not.

As the *isometric* device was shown to be particularly sensitive in Experiment 1, we conducted a planned comparison between *DLD* and *hinckley* for it at the shortest distance, revealing means of 1.57s with *DLD* and 2.1s with *hinckley* ($F_{1,15} = 7.4, p < .05$). Although further analysis is required, this suggests that *DLD*'s slow movement function can be effective in aiding slow precise control.

Subjective responses for workload (using the NASA Task Load Index [10]) favoured *DLD*. Table 2 summarises these values, showing that where significant differences were detected, they all favoured *DLD*. The table suggests that participants found *DLD* most beneficial with the *notch* device for which physical and temporal workloads were

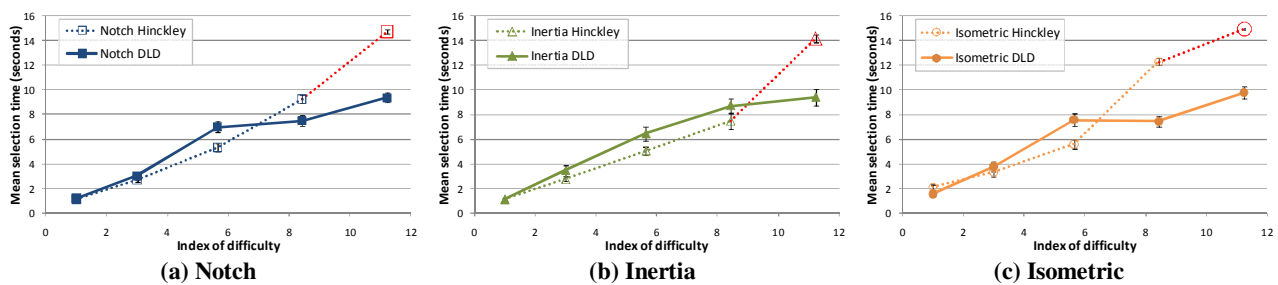


Figure 6. Mean selection time with the three devices with Hinckley and DLD Gain. The enlarged red final data point with Hinckley gain indicates that most trials exceeded the automatic 15 second time cap. Index of difficulty = $\log_2(\text{distance}/\text{width}+1)$.

particularly high with *hinckley*. Participants showed higher mean values for success when using *DLD* with all devices, and they rated it significantly higher for comfort. Despite these positive results, participants were near evenly split in their preferences for control.

Participants' comments reinforced these findings. One participant stated that when using *hinckley* with the *notch* device there was “*strain on finger, particularly index*” and “*very slow scrolling far away*”. Another stated that *hinckley* with the *isometric* device was “*Hard to manipulate*”. However, several participants commented on overshooting problems with *DLD*: “*kept over scrolling, hyper sensitive*” and “*didn't like how it jumped when I was close to target.*”

	Notch		Inertia		Isometric	
	DLD	Hinckley	DLD	Hinckley	DLD	Hinckley
Mental load	2.8(1.4)	2.2(1.1)	2.3(1.1)	2.1(1.1)	2.3(1.1)	2.0(1.1)
Physical load	2.5(1.0)	4.3(1.0)	2.2(1.0)	2.8(1.3)	2.4(1.1)	2.6(1.4)
Temporal load	2.4(0.7)	4.1(0.9)	2.6(0.9)	3.0(1.0)	2.8(0.9)	3.2(1.0)
Success	3.6(0.9)	2.6(0.9)	4.0(0.8)	3.4(0.8)	3.5(1.1)	2.4(0.8)
Hard work	2.9(0.8)	4.1(0.9)	2.4(0.8)	2.9(1.0)	3.1(0.9)	3.0(1.1)
Frustration	2.9(1.1)	3.2(1.2)	2.2(1.0)	2.4(0.9)	3.2(0.9)	2.9(1.0)
Comfort Pref.	16	0	13	4	12	3
Control Pref.	10	7	8	9	9	8

Table 2. Mean (st. dev.) NASA-TLX values (1= low, 5=high). Significant results are bold-italic, using Friedman tests for all but comfort and control preferences (which used χ^2 tests).

DLD's slight performance disadvantage for short distances and its substantial advantage for long distances (in this extremely long document) are indicative of a performance trade-off. When navigating long documents such as manuals, users are likely to need to move large distances (e.g., cross referencing the index), and these results show that for movements exceeding 10 pages (*notch* or *inertia*), *DLD* substantially outperforms *hinckley*. Part 2 examines *DLD*'s performance in shorter documents.

Part 2: DLD-gain performance across document length

Hypothesis 2 examines *DLD-gain*'s performance in shorter documents (10 and 100 pages). For expediency, experiment parts 2 and 3 were conducted using only the *inertia* device (which happened to have the smallest performance difference between *hinckley* and *DLD* in Part 1).

Participants completed trials at a subset of Part 1 distances (7, 49, and 343 lines) in a 600 line document and in a 6000 line document (recall that Part 1 used a 20000 line document). Trials were otherwise administered exactly as for Part 1, with the same progression through increasing movement distances and repetition of the two directions.

Part 2: Results

Acquisition time and overshoot data are analysed using a 3x3 RM-ANOVA for factors *distance* (7, 49, and 343) and *document length* (600, 6000, and 20000 lines).

Figure 7(a) summarises the acquisition time results, with *hinckley* data from Part 1 also shown for comparison. There was a significant main effect of *document length* ($F_{2,30} =$

20.3, $p < .001$), with means of 2.4, 2.8, and 3.7 for lengths 600, 6000 and 20000 respectively. Means for *distance* were also significant ($F_{2,30} = 37.9$, $p < .001$). Importantly, though, there was a significant *length* x *distance* interaction ($F_{4,60} = 14.3$, $p < .001$), caused by acquisition times increasing less steeply across distance in shorter documents. ANOVA of overshoots (summarised in Figure 7(b)) shows the same statistical effects, with overshooting much more prevalent in the 20,000 line document. We therefore accept **H2**.

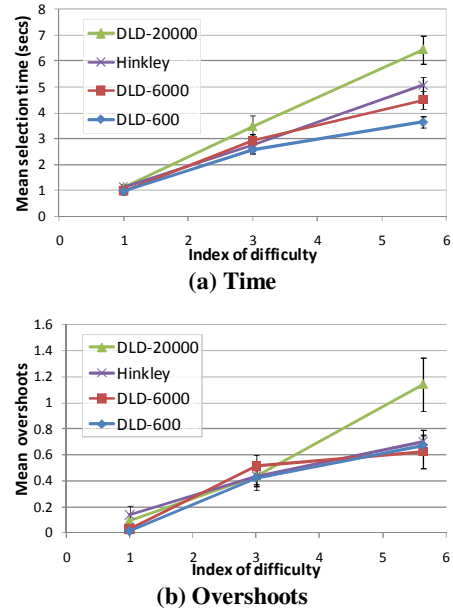


Figure 7. Part 2 results, including *hinckley* data from Part 1.

We conducted planned comparisons between *hinckley inertia* data from Part 1 with *DLD 600* and *DLD 6000* data from Part 2, using separate 2x3 RM-ANOVA for *gain* and *distance*. As suggested by Figure 7(a), in the 600 line document *DLD* (mean 2.4s) was significantly faster than *hinckley-inertia* (mean 3.0s): ($F_{1,15} = 20.5$, $p < .001$). There was also a significant *gain*x*distance* interaction ($F_{2,30} = 21.9$, $p < .001$) due to performance with *hinckley-inertia* deteriorating more rapidly across distance than *DLD*. Comparison of *DLD* performance in the 6000 line document with *hinckley-inertia* shows no main effect of *gain*, but a significant *gain*x*distance* interaction ($F_{2,30} = 4.3$, $p < .001$). Analysis of overshoots showed no significant effects involving *gain*.

These results suggest that *DLD-gain* outperforms *Hinckley gain* with the *inertia* wheel in a ten page document. This supports the explanation that *DLD*'s slightly worse performance than *Hinckley gain* in middle-distance scrolling in Experiment 2 Part 1 was due to the performance trade-off in allowing rapid extremely long distance scrolling at the cost of ‘twitchy’ performance in

middle distance scrolling. Experimental concerns with this assessment are presented in the Discussion.

Part 3: DLD-gain performance with document switching

Hypothesis 3 is motivated by the risk that performance may be adversely affected when rapidly switching between documents of different lengths (which will cause different gain behaviours with DLD). The change in gain behaviour is analogous to switching between different mouse gain settings, which temporarily disrupts pointing performance.

Participants completed two repetitions of bidirectional acquisitions with the following distance and document length tuples, in the following order: (49, 20000), (343, 600), (16807, 20000), (7, 600). This order was used to maximise abrupt transitions between document lengths, and therefore maximise changes in DLD-gain behaviour. To allow a comparison between scrolling times when working within one document and when switching between document lengths, we compare data from Part 3 (which we termed ‘switching’ trials) with data from the same distance and document length combinations in Experiment Parts 1 and 2 (‘non-switching’ trials).

Part 3: Results

Acquisition time data are analysed using a 2x4 RM-ANOVA for factors *task-type* and *distance-length*. The levels of *task-type* are *non-switching* and *switching*, with *non-switching* data extracted from the equivalent conditions in Parts 1 and 2 (where participants used the *inertia* device for the same distance and length conditions is a constant document length). The levels of *distance-length* were (7,600), (49, 20000), (343, 600) and (16807, 20000).

Figure 8 summarises the results. There were no significant negative effects of switching between different document lengths. The mean acquisition time for *non-switching* trials (4.4s) was similar to that for *switching* trials (4.1s): $F_{1,15} = 2.48$, $p = .14$. There was a marginal *task-type*x*distance-length* interaction ($F_{3,45} = 2.49$, $p = .07$), but as Figure 8 shows, this is due to *switching* being faster than *non-switching* trials at the longest distance-document combination, which (although not significant) is likely due to participants becoming more familiar with the inertia device later in the experiment. We therefore accept **H3**.

DISCUSSION

The results of Experiment 2 show that DLD-gain allows faster scrolling in extremely long documents than existing state-of-the-art gain functions, that its performance is faster still in more typical document lengths, and that performance is not harmed by switching between documents of different lengths. Subjective preferences were also favourable. These are good results, which suggest that the utility of scrolling devices such as scroll-wheels can extend far beyond their commonplace restriction to short-distance movement. The following paragraphs describe challenges to general use of DLD-gain, and areas for further work.

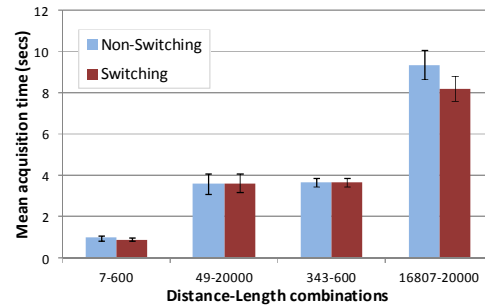


Figure 8. Part 3 results: effects of switching between different document lengths (Non-Switching data is from Parts 1 and 2).

Experimental limitations for further work

These current results provide a strong foundation for further empirical analysis of DLD-gain’s performance. We will pursue three main areas in further work.

First, we will compare DLD-gain with previous gain functions in typical document lengths. Experiment 2 Part 2 demonstrated that performance with DLD-gain improves in shorter documents, but we also compared DLD results from Part 2 with Hinckley gain results from Experiment Part 1. This comparison is reasonable because Hinckley’s gain functions are not affected by document length, but there are risks of learning effects and of task prompting effects (the scroll trough mark is smaller in a long document). We are confident that direct comparisons with short documents will confirm the results. Our second main area for further work is to conduct these comparisons with multiple devices.

Third, an important part of the motivation for DLD-gain was to enable scrolling devices to support the full range of scrolling activities, regardless of distance. We think DLD-gain achieves this objective, but in future work we will seek empirical evidence demonstrating the use of DLD-gain enhanced devices in favour of dragging the scroll thumb.

Application gain, rather than device

DLD-gain requires information about document length and viewport size that is unavailable to device driver software. It is most likely that DLD-gain would be implemented by particular applications or interface toolkits. This raises three concerns: first, how could application designers calibrate their software to work with many different devices; second, how would users adapt to inconsistent scrolling across applications; and third, how does DLD-gain interact with different device gain settings? For calibration parameters, without changes to driver protocols, the application would need to support a table lookup dependent on automatic device detection or user-specification of their main device. The second issue of inconsistent scrolling behaviour seems relatively minor as many applications already support scrolling enhancements that result in varied performance. Experiment 2 Part 3 also provided preliminary evidence that users are robust to sudden large changes in scrolling gain behaviour. The third issue of interaction between

device driver gain and DLD-gain is interesting and we intend to conduct further studies using the method of Casiez and Roussel [6] to examine and manipulate device gain.

Comparison with other techniques

Finally, many research systems are known to improve scrolling performance, with several using some form of zooming output effect (e.g., OrthoZoom [3], SDAZ [11], and SFT [8]) or oscillatory input gestures (e.g., Cyclopan [16] and the virtual scroll ring [17]). DLD-gain is different in that it maintains traditional mechanisms for scrolling input and output. However, empirical comparison with these and other techniques remains interesting.

CONCLUSIONS

Specific devices for scrolling, such as scrollwheels, are provided on most pointing devices, but there has been surprisingly little publicly accessible research into improving their performance. Consequently, these devices have limited applicability, and in particular, they become inefficient for long distance scrolling. We presented three gain functions, collectively called document-length-dependent gain, that improve performance with scrolling devices. They do so by making the maximum attainable gain level a function of document length, which allows long documents to be traversed quickly. Experimental results demonstrated that the technique improves performance and preferences over existing gain methods in a variety of document lengths and navigation distances.

ACKNOWLEDGEMENTS

Thanks to the experimental participants and the paper reviewers. This work was supported by a Royal Society of New Zealand Marsden Grant 10-UOC-020.

REFERENCES

1. Abowd, G. and Beale, R. Users, systems and interfaces: A Unifying Framework for Interaction. in *People and computers VI. Proc. British HCI '91*, 20, 1991, 73-87.
2. Aliakseyeu, D., Irani, P., Lucero, A. and Subramanian, S. Multi-flick: An evaluation of flick-based scrolling techniques for pen interfaces. in *Proc. CHI'08*, ACM Press, (2008), 1689-1698.
3. Appert, C. and Fekete, J. OrthoZoom Scroller: 1D Mutli-scale navigation. in *Proc. CHI'06*, ACM Press, (2006), 21-30.
4. Buxton, W. Lexical and pragmatic considerations of input structures. *SIGGRAPH Computer Graphics* 17, 1 (1983), 31-37.
5. Card, S., Mackinlay, J.D. and Robertson, G. A morphological analysis of the design space of input devices. *ACM Transactions on Information Systems* 9, 2 (1991), 99-122.
6. Casiez, G. and Roussel, N. No more Bricolage! Methods and Tools to Characterize, Replicate and Compare Pointing Transfer Functions. in *Proc. UIST '11*, ACM Press, (2011).
7. Casiez, G., Vogel, D., Balakrishnan, R. and Cockburn, A. The Impact of Control-Display Gain on User Performance in Pointing Tasks. *Human-Computer Interaction* 23, 3 (2008), 215-250.
8. Cockburn, A., Gutwin, C. and Alexander, J. Faster Document Navigation with Space-Filling Thumbnails. in *Proc CHI '06*, ACM Press, (2006), 1-10.
9. Cockburn, A., Savage, J. and Wallace, A. Tuning and Testing Scrolling Interfaces that Automatically Zoom. in *Proc. CHI '05*, ACM Press, (2005), 71-80.
10. Hart, S. and Staveland, L. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. in Hancock, P. and Meshkati, N. eds. *Human Mental Workload*, 1988, 139-183.
11. Hinckley, K., Cutrell, E., Bathiche, S. and Muss, T. Quantitative Analysis of Scrolling Techniques. in *Proc. CHI '02*, ACM Press, (2002), 65-72.
12. Hinckley, K. and Cuttrell, E. Distance-based accelerated scrolling. US Patent: 7,173,637 B1, 2007.
13. Igarashi, T. and Hinckley, K. Speed-dependent Automatic Zooming for Browsing Large Documents. in *Proc. UIST '00*, ACM Press, (2000), 139-148.
14. Ishak, E. and Feiner, S. Content-aware scrolling. in *Proc. UIST '06*, ACM Press, (2006), 155-158.
15. Jacob, R., Sibert, L., McFarlane, D. and Mullen, M. Integrality and separability of input devices. *ACM Transactions on Human-Compueter Interaction*, 1, 1 (1994), 3-26.
16. Malacria, S., Lecolinet, E. and Guiard, Y. Clutch-free panning and integrated pan-zoom control on touch-sensitive surfaces: the cyclostar approach. in *Proc. CHI '10*, ACM Press, (2010), 2615-2624.
17. Moscovich, T. and Hughes, J. Navigating documents with the virtual scroll ring. in *Proc. UIST '04*, ACM Press, (2004), 57-60.
18. Wobbrock, J., Findlater, L., Gergle, D. and Higgins, J. The Aligned Rank Transform for Nonparametric Factorial Analyses Using Only ANOVA Procedures. in *Proc. CHI '11*, ACM Press, (2011), 143-146.
19. Zhai, S., Smith, B. and Selker, T. Improving Browsing Performance: A Study of Four Input Devices for Scrolling and Pointing Tasks. in *Proc. INTERACT'97*, (1997), 286-292.