User Adaptability to System Delay

Brian D. Hall
University of Wisconsin-Stevens Point
Stevens Point, WI 54481, USA
brian.hall@uwsp.edu

CHI'16 Extended Abstracts, May 07-12, 2016, San Jose, CA, USA
ACM 978-1-4503-4082-3/16/05.
http://dx.doi.org/10.1145/2851581.2890381

Abstract
This double-blind, controlled, counter-balanced experiment examines the effects of system delay on users. Sixty-one participants completed 8 computerized tasks under 4 varied levels of time delay (from 500ms to 2000ms), with time and accuracy recorded. Participants rapidly adapted to system delay, committing neither errors of omission nor commission more frequently due to slowed system response. Time to finish tasks increased with system delay, but users were only slowed down by approximately as much time as was introduced by system delay itself. Implications for estimating the potential value of reduced system delay are discussed, as well as study limitations and suggestions for future work.

Author Keywords
System delay; response time; latency; user experience; human error; value estimation

ACM Classification Keywords
H.5.2 User Interfaces

Introduction
When users have a job to do and cannot just walk off because a computer is slow, how will system delay affect work? Will tasks take longer, and if so, will they only be slowed as much as they are directly delayed by the system? If system delay causes user error, what kind of errors will users make?

There may also be a tradeoff between system speed and user accuracy. Improvements in one metric, with equal and opposite deterioration in another, would be cause for concern. To be able to estimate any of these factors, allow improved organizational decisions, or guide the focus of future system designers, we need to understand more about how human work is affected by system delay.

Background and Previous Work

Concern over the impacts of system delay (at times referred to as response time or latency) on many factors of usability pre-dates the field of HCI, with perhaps the most cited work being Miller's 1968 paper [3]. Since then many studies have been conducted on topics ranging from command-line text entry to word processing and photo editing [5], robotic control systems [1], augmented reality [2], and even remote-controlled space-flight [4]. Results have been mixed, with some studies showing that some moderate delay can actually improve operator performance in a haptic input task [6], while other studies show that lower delays result in improvements until a general minimum threshold has been reached [4]. Other studies have shown that mild-to-moderate delays of no more than half a second had no discernible results on operator performance [1], while higher delays in other studies (ranging from 0 to 5 seconds) showed progressive increases in the time it took to perform the task [4].

If one is concerned with the sort of issues inherent in computer-mediated work-related tasks such as time, error rates, and the estimation of potential improvements, unfortunately existing studies have not generally focused on these unique combinations of concerns. Studies that have considered modern work-like tasks, such as use of Microsoft Word, have considered no concept of error, and users either completed an assigned task (such as inserting a picture into a document) or they failed to do so in the allotted time [5]. In reality most work tasks can be technically completed but performed poorly, yet this has not always been measured in previous studies.

There are also many potential confounding variables related to the interface itself. For instance, delayed scrolling actions in a remote desktop application may slow down users' ability to perform tasks [5]. This example of delay may not be applicable to a browser-based application where delays are more often caused by round-trip events like page reloads or asynchronous server responses, even while scrolling can remain relatively smooth.

Hypotheses

> *H1: System delay affects the time it takes to complete a task.*
> *H2: System delay affects the time it takes to review a task before submission.*
> *H3: System delay affects the time it takes to submit a task.*

H1-H3 examine the causal relationship between system delay and time on task. In H1, the time to complete a task is measured as the time between when the task is first presented to the user, to the time the user makes the final cell action. In H2, the time to review the task is the amount of time measured between when a user stops interacting with the task cells, to the time that the user officially submits the task. H3 concerns the time between the task first being presented to the user,

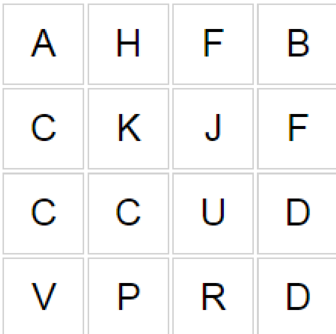

Figure 1: Blank miniature task, shown in the experimental instructions.

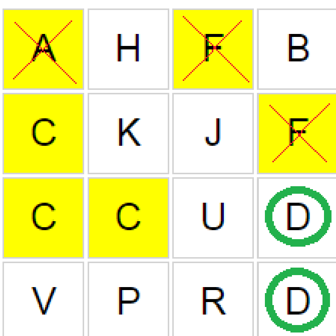

Figure 2: A "corrected" example shown in the experimental instructions. Cells with red crosses should not have been selected, and cells with green circles should have been highlighted.

to when the task is officially submitted by the user, and thus effectively combines the times considered in H1 and H2 into a single measurement.

> *H4: If the amount of system delay introduced into a task is removed from the amount of time it takes to complete a task, there will still be an effect of system delay.*
> *H5: If the amount of system delay introduced into a task is removed from the amount of time it takes to submit a task, there will still be an effect of system delay.*

H4 and H5 consider time on task that is adjusted to remove the time directly attributable to system delay.

> *H6: System delay affects the rate at which errors of commission occur.*
> *H7: System delay affects the rate at which errors of omission occur.*

H6 and H7 consider errors in a task irrespective of how long it took to complete the task itself. Error of commission is defined as selecting a cell that should not have been selected (incorrect cells selected). Error of omission is defined as failing to select a cell that should have been selected (correct cells missed).

Method

The goal of this experiment is to tie together and confirm aspects of previous research, improve understanding of specific time components of a task, improve understanding of certain types of task error, and to aid estimation of the potential impact for altering system delay. The experiment was designed with a focus on scientific rigor, in the hopes that any findings would prove both reliable and replicable. Extensive efforts were made to control against potentially confounding variables ranging from differences in hardware and software settings to even minor artifacts related to user interface design, such as animation or color. Double-blindness was used to minimize the chance for experimenter effects or participant expectancy, with random assignment and counter-balancing used to further limit the possibility for ordering or participant learning effects from reducing interpretability of the results.

Procedure

The experimental task in this study was designed to be objective and clear so that anyone of average intelligence could understand it fully. Participants were given 8 similar tasks in all, which varied only in the letters and placement of correct squares. Each task was a 10x10 grid table, with each grid cell holding an upper-case English letter. Participants were to highlight cells that matched any neighboring cell horizontally or vertically. Tasks were self-paced.

To ensure understanding of the task each participant completed two training tasks with 100% accuracy. These training tasks were 5x5 grids, otherwise identical to the full-sized tasks. Refer to figure 1 and 2 for the 4x4 version used in the initial instructions for the tasks.

The experimental manipulation was system delay in responding to a user's input in the task. When a participant clicked on a cell the application would wait the amount randomly assigned delay before executing the command. The application discarded all user input until the first requested command was completed.

Delay Condition	Mean	Std. Dev.
Time to Complete		
500ms	61274.2	20125.280
1000ms	67541.5	18535.876
1500ms	78776.4	20311.929
2000ms	89208.6	18885.253
Time to Review		
500ms	18964.4	13568.346
1000ms	19635.0	15535.246
1500ms	19125.7	13525.567
2000ms	18405.2	10803.249
Time to Submit		
500ms	80238.6	24610.792
1000ms	87176.5	28276.079
1500ms	97902.1	26130.569
2000ms	107613.8	22767.286
Time to Complete Minus Delay		
500ms	51718.6	19877.026
1000ms	47884.2	17679.619
1500ms	48545.1	19104.490
2000ms	48439.7	17087.353
Time to Submit Minus Delay		
500ms	70683.0	24316.001
1000ms	67519.2	27438.709
1500ms	67670.8	24878.570
2000ms	66844.9	21179.606
Correct Cells Missed		
500ms	2.0	1.740
1000ms	1.7	1.577
1500ms	1.5	1.415
2000ms	1.9	1.695
Incorrect Cells Selected		
500ms	0.0	0.064
1000ms	0.0	0.142
1500ms	0.0	0.090
2000ms	0.0	0.064

Figure 3: Descriptive statistics for all levels of system delay, categorized by dependent variable. Note that N=61 for all rows.

Apparatus

The experimental application was specially designed for the purpose of this study, and ran in full-screen mode on desktop computers in a reserved University computer lab. No task switching was possible, there were no menu or navigation options presented to the user, and only left-clicking with the mouse would perform an action of any kind.

Participants

Sixty-one participants were recruited from an online pool of undergraduate students to take part in this study in exchange for research participation course credit. Students were only required to be at least 18 years of age (N=61). Double-blind procedures were used, as all participants were randomly assigned into counter-balanced task orders and task-delay condition pairings by the experimental application and were unknown to the experimenter.

Results

All 7 research hypotheses were tested at once with a repeated measures multivariate analysis of variance (MANOVA). The independent variable was system delay, which had four levels, as well as 7 dependent variables. Refer to Figure 2 for a full listing with descriptive statistics. The multivariate test was found to be significant with F (15, 46) = 665.648, $p < 0.001$. Consequently, pairwise comparisons were calculated for all dependent variables utilizing a Šídák correction for multiple comparisons to reduce type I errors.

For H1 the time to complete was compared for each delay level. The difference in the 500ms condition and the 1000ms condition was not statistically significant ($p > .072$). However, all other comparisons of delay condition were found to be statistically significant ($p < .001$). We thus reject the null, and assert that increasing system delay (at least beyond 1000ms) increases the time it takes to complete a task.

For H2 the time to review before submitting the task was compared for each delay level. In no condition was any difference found to be significant (values of $p > .992$). We thus fail to reject the null, and conclude there is not enough evidence to indicate that system delay has any effect on the time it takes to review a task before submitting it.

For H3 the time to submit was compared for each delay level (see figure 4). Every comparison was found to be significant, with the 500ms condition compared to the 1000ms condition having $p < .01$ and all other cases $p < .001$. We thus reject the null, and assert that increases in system delay (from 500ms to 2000ms) increase the time it takes to submit a task.

For H4 the delay added to the task was subtracted from the time it took to complete the task, and this delay-adjusted completion time was compared for each delay level. In no condition was any difference found to be significant ($p > .500$). We thus fail to reject the null, and conclude there is not enough evidence that system delay has any effect on the time it takes to complete a task once we have adjusted for the delay added.

For H5 the delay added to the task was subtracted from the time it took to submit the task, and this delay-adjusted submission time was compared for each delay level. Just as with H4, no pairwise comparison was deemed statistically significant ($p > .200$). We thus fail to reject the null, and conclude there is not enough

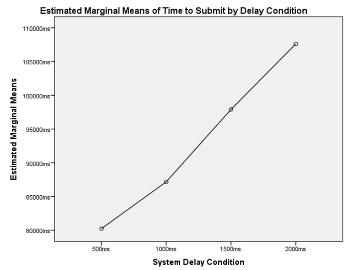

Figure 4: Means for time to submit tasks, without consideration for the amount of time added by system delay. Participants took longer to submit tasks as delay increased, but further analysis showed that this was attributable entirely to the amount of system delay added.

evidence that system delay has any effect on the time it takes to submit a task once we have adjusted for the delay added.

For H6 the amount of incorrect cells we turn away from time on task and instead consider errors of commission in the form of cells that the participants selected that were not valid to highlight. In this experiment this type of error was not regularly committed, with a mean of 0 across conditions ($p > .961$). We thus fail to reject the null, and conclude there is not enough evidence that system delay has any effect on errors of commission.

Finally, for H7 we examine errors of omission in the form of cells missed, which were cells that should have been highlighted but that participants did not select. In pairwise comparisons there were no significant differences, with $p > .100$ in all cases. We thus fail to reject the null, and conclude there is not enough evidence that system delay has any discernible effect on the occurrence of errors of omission.

Discussion

The time it takes to complete or submit a task under system delay initially seemed to agree with previous studies, which suggest that increasing delay slows down users. Unlike previously identified studies, in this experiment the amount of time attributable to system delay was measured in addition to total user time on task. Once we adjust the time participants spent on task to account for the amount of time they were made to wait by the system, we find no remaining effect of system delay.

That users are resilient and adaptive to system delay is supported by other findings, as well. The occurrence of both errors of commission and errors of omission show that participants adapted to the conditions and performed the same work regardless of the delays they were forced to endure. In the measurements of time to review the task before submission, no effect of system delay could be detected. We can safely say that reducing system delay in this experiment would have allowed participants to finish sooner only to the extent we reduced delay, while their accuracy would have remained unchanged.

This single experiment has a few findings to offer to the field of HCI. For one, error is a universal issue in systems that involve humans. The fundamental difference between taking an explicit action and not acting is well illustrated by this study; people are more likely to overlook things than act in error.

The strongest overall finding of this study is that improvements in system delay - such as reducing a 2 second delay to a half-second – does not increase accuracy in simple tasks. The best that can be expected from a small improvement to system delay is an equally small improvement to task completion time, with no further detectable effect. Organizations and practitioners should thus be wary of micro-optimizations that will require significant time and effort.

Limitations and Future Work

There are some aspects of this study and findings that suggest a direction for future research. First, these tasks were not intellectually demanding and were unlikely to cause a high cognitive load. Previous studies suggest that there may be an interaction between task difficulty and system delay, and this was not included as part of this experimental design.

This study also included no delays below 0.5 seconds or beyond 2 seconds, which for some especially low/high response applications may not be sufficiently instructive. All delays within a task were consistent, rather than the jittering and perceptually random latency that is common in congested network scenarios or with resource-intensive applications. All conditions in this study were "locking" in nature, effectively forcing users to wait on the system if they were to get anything done. System response was also confirmatory in nature, and did not provide unknown information to the user.

The research application created for this experiment collected an extensive amount of data beyond what was required to solely address the hypotheses, with more than 25,000 individual task events and high-resolution timestamps offering sub-millisecond accuracy. Exploratory analysis of this data could shed additional light on whether cells missed were similar across participants, visual search strategies used, factors related to within-subject predictors of error or speed, the relationship between time to complete and errors made, and many more. Any such findings would require follow-up studies to confirm.

Finally, future experiments could seek to extend instrumentation or even move study participation outside the lab. Gaze-tracking and recording of mouse movements could examine user attentiveness to the task and provide additional insight into why users overlook important information. Online "crowd sourcing" of research participation could also ensure general applicability of experimental results with people who may not work in a low-distraction environment.

References

1. Daniel Cannon and Mel Siegel. 2015. Perceived mental workload and operator performance of dexterous manipulators under time delay with master-slave interfaces. *2015 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, IEEE, 1–6. http://doi.org/10.1109/CIVEMSA.2015.7158613
2. Ming Li, Katrin Arning, Luisa Vervier, Martina Ziefle, and Leif Kobbelt. 2015. Influence of temporal delay and display update rate in an augmented reality application scenario. *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia - MUM '15*, ACM Press, 278–286. http://doi.org/10.1145/2836041.2836070
3. R.B. Miller. 1968. Response time in man-computer conversational transactions. *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, 267–277. http://doi.org/10.1145/1476589.1476628
4. Thomas B. Sheridan. 1993. Space teleoperation through time delay: Review and prognosis. *IEEE Transactions on Robotics and Automation* 9, 5: 592–606. http://doi.org/10.1109/70.258052
5. Brandon Taylor, Yoshihisa Abe, Anind Dey, Mahadev Satyanarayanan, Dan Siewiorek, and Asim Smailagic. 2015. Virtual Machines for Remote Computing : Measuring the User Experience. Carnegie Mellon University.
6. Sander Zuidhoek. 2005. Representations of space based on haptic input. Utrecht University.