Towards New Widgets to Reduce PC Power Consumption

Ross McLachlan
Stephen Brewster
Glasgow Interactive Systems Group
School of Computing Science
University of Glasgow
Glasgow, G12 8QQ
r.mclachlan.1@research.gla.ac.uk
stephen.brewster@glasgow.ac.uk
www.dcs.gla.ac.uk/~stephen

CHI 2012, May 5–10, 2012, Austin, TX, USA.
ACM 978-1-4503-1016-1/12/05.

Abstract
We present a study which describes the power consumption characteristics of a number of different interaction techniques on a desktop and laptop computer. In total, 8 interactions that can be used to carry out a single task (navigating a PDF document) were compared for power consumption across both a desktop and a laptop computer and across two different power saver settings. The results suggest that the power consumption of different interaction techniques for a single task vary significantly. Furthermore, the results suggest that a key factor in the power consumption of the interaction technique is the number of screen updates involved.

Keywords
Power consumption, widgets

ACM Classification Keywords
H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms
Design, Human Factors

Introduction

It is always desirable to find ways to reduce the power consumption of electronic devices. Power saving techniques for computing devices have tended to focus on optimising hardware or prolonging battery life on mobile devices, but rarely on optimising the user interface. In this paper we discuss a study which characterises the power consumption characteristics of a number of different interaction techniques to see if we can reduce power consumption by changing the interactions we use.

There have been several examples of new widgets developed to show the current power consumption of devices with the aim of helping users understand their own consumption and change their behaviour to consume less [1, 4]. Our approach is to change the way the interactions work to see if we can reduce power consumption in that way. This means that even if users do not change their behaviour power will still be saved.

Our longer term goal is the design of novel 'low power' interaction techniques. Devices currently use large screens and 'always on' connections which are convenient for the user but consume significant amounts of power. Different types of interactions consume different amounts of power but are suitable for different types of task. For example, audio displays might be cheaper than visual ones, so perhaps information could be read to the user rather than displayed on a screen. Gesture input might be cheaper than speech, and accelerometer-tracked gestures might be cheaper than camera-tracked ones. As our project progresses we will investigate the trade-offs between these different interactions in terms of power consumption and usability. However, in order to design such interactions we must first gain an understanding of how current interaction techniques consume power, which is the focus of this paper.

There has been some work on both characterising and developing energy efficient user interfaces in the context of battery powered devices. However, to our knowledge, the effect of user interface design on overall power consumption has not been investigated more generally for mains powered devices.

GUI Energy Characterisation

There have been a number of studies investigating the power consumption of the GUI, though they have been primarily concerned with battery optimisation on handheld devices. Vallerio *et al.* [5] developed and evaluated E2GUI, an energy efficient GUI platform for handheld devices. The platform was designed to reduce energy consumption by using low energy colour schemes and reducing the number of screen changes. For their three example applications, a text reader, a contact viewer and a calculator, they observed a 26.9% 45.2% and 16.4% reduction in energy consumption respectively compared to the standard implementations. The GUIs were also evaluated in terms of user preferences in the form of a questionnaire, the results of which suggested that the participants would be prepared to makes sacrifices in ease of learning and aesthetics of the GUI for improved energy efficiency (which, in this case, translated to prolonged battery life of their device).

There have also been proposals which aim to reduce the power consumption of the GUI by taking advantage of Organic Light Emitting Diode displays (OLEDs). OLEDs have different power consumption when displaying different colours [2] and so by either changing the colour characteristics of the GUI (as in [2]) or by dimming parts of the screen not currently in use (as in [3]), the power consumption of the mobile device can be reduced. This solution would work at the desktop too but would require existing screens to be replaced, which itself would be wasteful. We are interested in finding a software solution that could reduce power consumption which would be easy to roll out to many users.

Zhong *et al.* [6] carried out a complete energy categorisation of the GUI elements on three handheld devices. Their results suggested that there can be a significant difference in the power consumption of different interface widgets and interaction techniques. They observed that the same function can be implemented using different window types, or widgets, which often have different energy efficiencies (such as tabbed panels vs. scroll bars for browsing a long list) and that there are often constraints on the GUI platform that can cause widgets to be unnecessarily inefficient, such as built in animations on drop down menus. In particular, they found scrolling to be one of the most inefficient interactions, and suggest that it should be avoided, although they do not fully assess the available alternatives. They provide five recommendations based on their findings for creating more energy efficient GUIs: accelerate user interaction, minimise screen changes, avoid or minimise text input, reduce redundancy and do something while waiting for user input. We based our choice of scrollbar on their results as it would have the same kinds of problems on desktop GUIs.

Experimental Design

We designed an experiment to compare the power consumption of different interaction techniques that can be used to navigate through a PDF document. We choose to analyse this task for the following reasons: 1) document viewing typically involves scrolling, which was suggested by Zhong [6] to be one of the most inefficient and 2) in a typical PDF viewer application, there are many interaction techniques that accomplish the same task. The document used was an independent report on intellectual property and growth commissioned by UK government. This document was chosen as it is of a reasonable length (130 pages), is formatted with both text and graphics and is freely available[1].

We choose to take a black box approach and take the measurements using a widely available PDF Reader (Adobe Acrobat X Version 10.1.1). We used a Voltech PM1000+ Power Analyser to take the measurements. The computer being studied was connected in series with the power consumption monitor which itself was connected via USB to another computer running the logging software. Instantaneous power (in Watts) was sampled at a frequency of 1Hz for the full duration of the interaction. A measure of power consumption for each interaction was obtained by taking the average of the samples.

We took measurements on both a desktop (Dell Precision T3500 with Intel Xeon 2.67GHz processor and 4Gb RAM) and a laptop computer (Samsung SF311 with Intel Core i5 2.53Ghz processor and 4Gb RAM) both running Windows 7 Professional 64bit. The measurements were taken on the laptop when it was fully charged and plugged into the mains. We chose these two different devices to see how consumption varied,

[1] http://www.ipo.gov.uk/ipreview-finalreport.pdf (last accessed 06/01/2012)

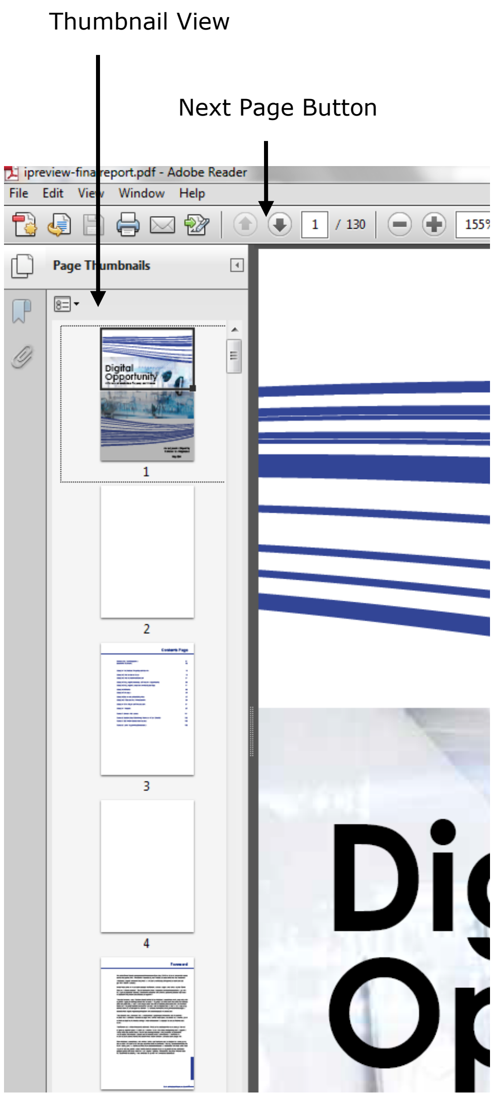

Figure 1: Screenshot showing the Thumbnail view and the Next Page Button

thinking that the laptop may already be more power optimised than the desktop PC. In order to control the effects on power consumption of any power saving mechanisms employed by the operating system, we took measurements with both power saver mode on (all available power saving techniques were enabled) and power saver mode off (no power saving techniques were enabled).

Measurements were taken of the power consumption of seven different interactions that can be used in order to complete the task `navigate to the end of the document'. We recognise that the task may be somewhat unrealistic, though it was simple enough that consistent measurements could be taken by a single researcher performing the interactions. Each interaction was carried out 12 times under each condition and an average power consumption figure calculated. The following interactions were used in the experiment:

- *Auto Scroll*: This was the Auto Scroll function built into Adobe Acrobat. It was activated by pressing Shift-Ctrl-H and was configured to run at full speed.

- *Down Arrow Key*: The down arrow key on the physical keyboard. The keyboard was connected via USB on the desktop computer and was built into the laptop. The key was held down which caused the document to scroll at a similar speed as the auto scroll function above.

- *Right Arrow Key*: The right arrow key on the physical keyboard (as above). Pressing the key once would advance the document one whole page. This action was repeated until the task was complete.

- *Mouse Wheel*: This was the wheel on a Dell USB mouse connected to the computer by USB. The same mouse was used for the desktop and laptop computers. Dragging the wheel towards the user caused the document to scroll downwards. This action was repeated until the task was completed.

- *Mouse Drag*: This action was carried out by using the same mouse as described above to click the thumbwheel on the scroll bar and drag it downwards until the end of the document was reached.

- *Thumbnail Navigation*: In Acrobat Reader there is a 'Thumbnail View' mode that displays, in a scroll pane on the left hand side of the window, a vertical stack containing a thumbnail representation of each page. This interaction was carried out by loading the thumbnail view (by selecting it with the mouse from the icon on the left of the screen) and clicking the thumbwheel on the scroll bar and dragging it to the bottom. Once the thumbnail of the last page of the document had been reached, it was clicked in order to load it onto the main viewing panel (See Figure 1).

The independent variables in the experiment were the Machine Type (Laptop or Desktop), the Power Saving Mode (On or Off) and Interaction. The dependant variable was Average Power Consumption.

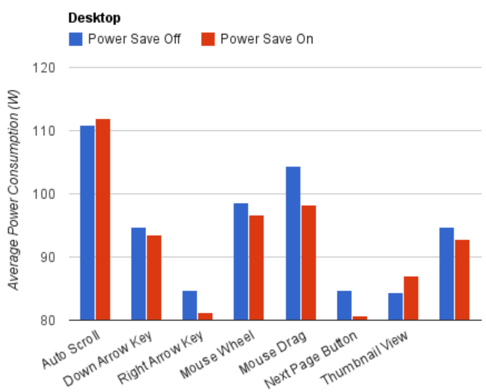

Figure 2: Average Power Consumption values for the Desktop Condition.

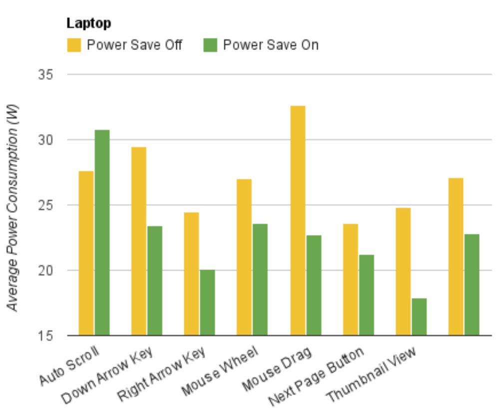

Figure 3: Average Power Consumption values for the Laptop Condition.

Results

A 3-way within subjects ANOVA was carried out on the power consumption measurements. The main effects for Machine Type, Power Saving Mode and Interaction were all significant ($p < 0.001$) as were the interactions between Machine Type x Power Mode ($p < 0.001$), Machine Type x Interaction ($p < 0.001$), Power Mode x Interaction ($p < 0.001$) and Machine Type x Power Mode x Interaction ($p < 0.001$).

Figures 1 & 2 show the overall averages of the power consumption values for each interaction. Across all Interactions, the laptop consumed less power than the desktop computer. Furthermore, the Power Saver mode was more effective in saving power in the laptop condition (Power Save On M=22.9, SD=4, Power Save Off M=27.1, SD=3.2) than in the Desktop condition (Power Save On M=92.8,SD=10.9, Power Save Off M=94.7,SD=10.6).

Post hoc pairwise comparisons with Bonferroni correction revealed a significant difference between most of the Interactions across all conditions suggesting that there can be great variability in the power consumption of even the same interaction across different machines and power saving modes. The lowest power consuming interactions across all machine types and power modes were the Down Arrow Key and Mouse Wheel interactions, the Right Arrow Key and Next Page Button interactions, the Right Arrow Key and Thumbnail View interactions and the Next Page Button and Thumbnail View interactions.

Discussion

The results provide an interesting insight as to what the key factors may be in the power consumption characteristics of document navigation interaction techniques. The Right Arrow Key, Next Page Button and Thumbnail View were the three lowest power consuming interactions in the experiment. All three of these interactions share the property of minimising the frequency of screen updates. In the case of both 'Right Arrow Key' and 'Next Page Button' by traversing whole pages of the document at once or, in the case of 'Thumbnail View', by reducing the magnitude of the screen update by only scrolling through icon representations of the pages. The other interactions in the experiment in some way involved scrolling through entire pages with, presumably, different levels of processing required in the background to render each page with all its text and graphics. This was typified in the case of the Auto Scroll function, which traversed the document by scrolling through each entire page.

The results obtained here suggest that there can be a significant difference in the power consumption of different interaction techniques for performance in an identical task. However, by reducing the number and frequency of screen updates in an interaction, power consumption can be reduced. We understand that simply because an interaction technique consume less power does not necessarily make it superior. However, from this, we do propose that power consumption could become a factor in future HCI experiments, where it is evaluated alongside more traditional usability measures for new interaction techniques.

Future work will focus on generalising the results presented here for different interactions over different applications. Furthermore, future work will also aim to develop alternative interaction techniques which have low power consumption and to evaluate the usability and user acceptance of those techniques. For example, it might be possible to reduce the number of screen updates, and thus power consumption, of a PDF reader by modifying it to display less of the document during scrolling. For example, the reader might only display section headings and figure captions, while hiding the main text. This would that less screen updates would have to take place, while still making it possible to navigate the document.

Conclusions

In this paper we present a study which characterises the power consumption of a number of different interaction techniques. In total, 8 interactions that can be used to carry out a single task (navigating to the end of a PDF document) were compared in terms of power use across both a desktop and a laptop computer and across two different power saver settings. The results suggest that the power consumption of different interaction techniques for a single task vary significantly. Furthermore, the results suggest that a key factor in the power consumption of the interaction technique is the number of screen updates involved. We found that techniques which enabled the user to traverse large section of the document (such as a next page button), and thus reduced the number of screen updates, consumed less power than scrolling based techniques.

References

[1] Chetty, M., Brush, A.J.., Meyers, B.R. and Johns, P. 2009. It's not easy being green: understanding home computer power management. *Proceedings of ACM CHI 2009* (Boston, MA, USA, 2009).

[2] Dong, M., Choi, Y.S.K. and Zhong, L. 2009. Power modeling of graphical user interfaces on OLED displays. *Design Automation Conference, 2009. DAC'09. 46th ACM/IEEE* (2009), 652–657.

[3] Harter, T., Vroegindeweij, S., Geelhoed, E., Manahan, M. and Ranganathan, P. 2004. Energy-aware user interfaces: an evaluation of user acceptance. *Proceedings of the SIGCHI conference on Human factors in computing systems* (2004), 199–206.

[4] Reitberger, W. and Tscheligi, M. 2007. Persuasion as an ingredient of societal interfaces. *Interactions*. 14, (2007), 41-43.

[5] Vallerio, K., Zhong, L. and Jha, N.K. 2006. Energy-efficient graphical user interface design. *IEEE Transactions on Mobile Computing*. (2006), 846–859.

[6] Zhong, L. and Jha, N.K. 2003. Graphical user interface energy characterization for handheld computers. *Proceedings of the 2003 international conference on Compilers, architecture and synthesis for embedded systems* (2003), 232–242.