Why it's quick to be square: Modelling new and existing hierarchical menu designs

Conference Paper · April 2010

DOI: 10.1145/1753326.1753534 · Source: DBLP

4 authors, including:

Carl Gutwin
University of Saskatchewan

Pourang Irani
University of British Columbia - Okanagan

Why it's Quick to be Square: Modelling New and Existing Hierarchical Menu Designs

David Ahlström[1], Andy Cockburn[2], Carl Gutwin[3], Pourang Irani[4]
[1]Dept. of Informatics Systems, Klagenfurt University, Austria, david@isys.uni-klu.ac.at
[2]Dept. of Computer Science, University of Canterbury, New Zealand, andy@cosc.canterbury.ac.nz
[3]Dept. of Computer Science, University of Saskatchewan, Canada, gutwin@cs.usask.ca
[4]Dept. of Computer Science, University of Manitoba, Canada, irani@cs.umanitoba.ca

ABSTRACT
We consider different hierarchical menu and toolbar-like interface designs from a theoretical perspective and show how a model based on visual search time, pointing time, decision time and expertise development can assist in understanding and predicting interaction performance. Three hierarchical menus designs are modelled – a traditional pull-down menu, a pie menu and a novel Square Menu with its items arranged in a grid – and the predictions are validated in an empirical study. The model correctly predicts the relative performance of the designs – both the eventual dominance of Square Menus compared to traditional and pie designs and a performance crossover as users gain experience. Our work shows the value of modelling in HCI design, provides new insights about performance with different hierarchical menu designs, and demonstrates a new high-performance menu type.

Author Keywords
Menus, hierarchical menus, performance models.

ACM Classification Keywords
H5.2. [User Interfaces]: Interaction Styles.

General Terms
Human Factors.

INTRODUCTION
User interface designers have many alternatives for providing access to commands in their systems. Menus and toolbars are primary mechanisms for selecting commands, and consequently researchers have proposed many designs to improve their performance: traditional menus use a linear item organisation, while toolbars use two-dimensional layouts; the 'Ribbon' interface (MS Office 2007) combines these two approaches (Figure 1); and radial 'pie' menus further extend the range of alternatives available to designers. While these alternatives provide designers with the flexibility to best match their users' needs, they also complicate the design process because it is hard to predict how different alternatives will perform. Would traditional menus, toolbars, pie menus or ribbons be fastest, and is comparative performance influenced by user experience, the structural layout of the menus, or by the hierarchical organisation? Designers are most likely to answer these questions using intuition or with empirical tests. However intuitions can often be wrong, and the time-consuming nature of empiricism limits the range of alternatives that can be considered. Theoretical performance models, in contrast, can be quickly and easily deployed.

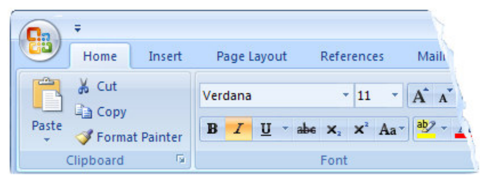

Figure 1. The 'Ribbon' replaces menus with tabbed toolbars.

A recent model of menu selection time [14], called the Search, Decision and Pointing (SDP) model, provides a synthesis and extension to Card *et al.*'s [11] seminal work on Keystroke Level Models. It predicts performance based on the time needed to either search or decide about an item, followed by the pointing time needed to select it. SDP proposes that novices rely on visual search to find targets (a linear function of the number of items) while experts can decide about their location (a logarithmic function, based on the Hick-Hyman Law of choice reaction time [22, 24]). The model therefore accounts for novice-to-expert transitions with different designs as users learn item locations. Although the model is promising, it is limited in that it has only been applied to linear menus (where items are below one another) and it has not been tested with menu hierarchies.

In this paper, we test the SDP model with hierarchical menus and apply it to a broader range of menu behaviours than previously investigated. We extract a crucial performance principle from the model – that experts spend proportionally far more time in the pointing phase of menu selection than novices. This suggests that menu designs for experts should focus on reducing pointing time. Using this principle, we identify two candidate designs that should perform well: radial pie menus, which have rapid but unusual pointing properties; and a novel menu type called

CHI 2010, April 10–15, 2010, Atlanta, Georgia, USA.

Square Menus that we designed to reduce pointing time without dramatically altering the shape of the menus.

We used the SDP model to predict performance with hierarchical pie, Square, and traditional linear menus. After calibrating the models, our predictions indicated that Square Menus would perform best for experts, and that there would be a performance crossover between these designs and traditional menus as users became more familiar with the menu items. An empirical validation study confirmed both of these predictions: Square Menus were significantly faster for experts than either traditional or pie menus, and the initial advantage for traditional menus disappeared by the second block of trials. In addition, our studies identify performance characteristics of pie menus that have not previously been reported (despite the long history of research into this design).

Our paper makes five specific contributions:

- Testing the SDP model with hierarchical menus;
- Demonstration of the model's value in creating new designs and comparing different approaches;
- Introduction of the novel Square Menus design;
- New performance characteristics of radial pointing;
- Empirical evidence that Square Menus are significantly faster than pie and traditional menus.

Our experiences show that performance models can have value both in helping designers understand underlying principles of performance and in helping them identify and compare different design alternatives.

RELATED WORK

Menu Designs

There has been extensive research on improving interaction with menus. In general, there are four underlying objectives for the improvements, reviewed below: reducing Fitts' Law [17] target acquisition time, aiding target identification, reducing Steering Law [1] time in hierarchical menus, and improving menu shortcuts. Many of these techniques draw on the non-uniform nature of command use (i.e., a few commands are used often, and many are seldom or never used [15, 18, 20]).

Reducing Fitts' Law target acquisition time

Fitts' Law [17] is a robust rule of human movement, predicting that movement time (*MT*) of a limb (or cursor) to a target follows the formula $MT = a + b \times log_2(A / W + 1)$, where A is the movement distance, W is the target width, and a and b are empirically determined intercept and slope constants. The logarithmic term is called the movement 'index of difficulty'. Much research in HCI has been focused on reducing pointing time in various contexts (see [6] for a review), including menus.

Split menus [32] are widely used in commercial software. They reduce target distance for the most frequently or recently used items by moving or copying them into a special region at the top of the menu. Studies have shown that frequency-based split menus provide greater menu stability and therefore better support users in developing expertise [14] than recency-based split menus. However, the overall merit of splits is not clear, and some unfavourable experimental results have been seen [14, 15].

Instead of reducing pointing distance, morphing menus [14] increase the width of frequently used items. Like split menus, their performance benefits over traditional menus is not established. Bubbling menus [36] also increase the effective target width of frequent items, but do so by changing cursor size. This introduces the need for a mode partition, with one menu side disabling bubbles to allow selection of infrequent items, while the other side enables the bubble cursor. Experiments suggest that Bubbling menus are effective in hierarchical menu selections [36].

Radial 'pie' menus [9] and gestural marking menus [26, 27] arrange targets in a circular layout around the cursor (see Figure 2). Targets are selected by dragging in their direction. They offer several interaction benefits: all targets are accessible with only a small cursor movement, and items can be selected with rapid gestures before the menu is displayed. Many studies have evaluated various forms of radial menus and proposed iterative refinements. Callahan et al. [9] demonstrated that error rates with pie menus remain low with fewer than eight items per menu, and Kurtenbach and Buxton [26] showed that users can achieve better than 90% accuracy using compound gestural strokes to navigate through two-level hierarchies of 64 items. Several radial command systems allow broader structures by attending to the position and orientation of strokes [38] or their curvature [4]. Finally, radial Wave menus [5] use expanding concentric circles to aid novices.

The empirical and design work on radial menus is impressive, providing rich characterisations of use. Surprisingly, however, there has been no direct performance comparison between hierarchical pie menus and the traditional ones used in current systems.

Steering Law time in hierarchical menus

The Steering Law [1] predicts that the time to move the cursor through a constrained movement 'tunnel' is a linear function of the ratio of tunnel length to width. Steering is pertinent in many hierarchical menu selections that involve

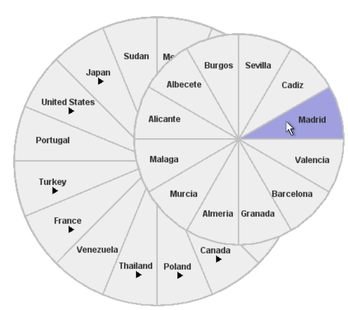

Figure 2. A hierarchical pie menu, as used in Experiment 2.

moving to a particular cascading menu item, then accurately steering across it to enter a lower level menu.

Several techniques have tried to improve performance by reducing horizontal steering distance. Kobayashi and Igarashi [25] describe a technique that reduces the horizontal tunnel length by posting cascading menus under the cursor when the user begins to drag across the menu. Their study showed performance improvement over traditional menus. Ahlström *et al.* [3] described two techniques that effectively reduce steering distance through cursor warping: 'Jumping' menus instantaneously warp the cursor into the middle of the hierarchical menu when the parent is clicked, while 'Force enhanced' menus gradually displace the cursor towards cascades. Both techniques improved selection times, although Jumping menus caused many more errors.

Another approach is to increase the width of the steering tunnel. EMUs [12] enlarge the activation area that triggers cascade menu posting, reducing both the need for sharp corners in the cursor trajectory and the need for menu posting/unposting timeout delays. Adaptive activation area menus [34] are shown to improve on this approach by dynamically calculating an enlarged activation area that contains the triangular region from the cursor's location to the start and end of the next level menu.

Improving menu target identification

Several menu systems have investigated methods that help users lock on to targets. Findlater *et al.* [16] introduced 'ephemeral adaptation', which immediately displays probable menu targets, while slightly delaying and fading in other items. Results showed that the technique improved performance over traditional menus and colour highlighting by successfully capitalising on human factors of visual processing, in which items with abrupt onset are processed first. Other researchers have examined audio [38] and haptic feedback [31] to help users identify targets when their visual attention is directed elsewhere.

Improving menu shortcuts

Finally, researchers have improved methods for issuing and recalling menu shortcuts. Marking menus support a natural transition from novice to expert performance by allowing novices to visually scan contents, while experts use rapid directional gestures [26, 27]. Grossman *et al.* [19] devised methods to promote the use and retention of keyboard shortcuts, with good results.

Menu Models

In addition to the strong empirical and design work on menus, a variety of theories have been proposed to model menu use. Novices' visual search for menu items has been modelled as traversing over candidate items randomly [10], linearly [8, 23, 29], and in parallel [23], but eye tracking data support a predominantly top-to-bottom search [8] in support of a 'maximally efficient foveal sweep' [23]. Several studies agree that novices' search time is a linear function of menu length [14, 23, 32]. Expert menu use has been modelled in many ways, including Fitts/Steering Law [2, 33], KLM [28], GOMS and ACT-R [7, 33], and Huffman codes [35]. Our focus, however, is on the recent 'Search, Decision and Pointing' model [14] of novice *and* expert use.

The Search, Decision, and Pointing (SDP) Model

The SDP model [14] integrates several low level models to form performance predictions with single level linear menus (where items are shown in a single vertical column) across experience levels, summarised below.

The average time T_{avg} to select items in a menu is calculated as the probabilistic sum of times for its constituent entries (where p_i is the probability of item i):

$$T_{avg} = \sum_{i=1}^{n} p_i T_i \qquad (1)$$

Item selection time T_i is calculated as the sum of the two sub-tasks that involve first finding the item (T_{dsi}) and then acquiring it (T_{pi}).

$$T_i = T_{dsi} + T_{pi} \qquad (2)$$

For menus using traditional cursor movement for target acquisition, pointing time T_{pi} is calculated with Fitts' Law:

$$T_{pi} = a_p + b_p \log_2(A_i / W_i + 1) \qquad (3)$$

Unusual designs, however, such as radial pointing, may have different pointing characteristics (see Experiment 1).

The time to find the item (T_{dsi}) depends on the user's level of expertise (e_i from 0 to 1): novices visually search, experts decide about location, and intermediates do some of both:

$$T_{dsi} = (1 - e_i) T_{vsi} + e_i T_{hhi} \qquad (4)$$

Visual search time (T_{vsi}) is a linear function of menu length (n), where a_{vs} and b_{vs} are empirically derived intercept and slope values:

$$T_{vsi} = b_{vs} n + a_{vs} \qquad (5)$$

Expert decision time (T_{hhi}) is calculated using the Hick-Hyman Law of choice reaction time, where a_{hhi} and b_{hhi} are empirically-derived intercept and slope constants, and H is termed the information entropy of the decision:

$$T_{hhi} = b_{hh} H + a_{hh}, \text{ where } H = \log_2(1/p_i) \qquad (6)$$

Item expertise *(e_i)* is calculated as a power law of practice [30] dependent on previous experience selecting the item (trials, t_i) and on interface learnability (L, from 0 to 1):

$$e_i = L \times (1 - 1/t_i) \qquad (7)$$

A menu's learnability depends on its spatial stability – designs that frequently move items (e.g., split menus that move items rather than copy them) will hinder users' ability to learn locations.

Although this series of equations may appear unwieldy, they are very simply implemented in a spreadsheet. In a spatially stable interface design, the only parameters that require calibration are the a and b constants for Fitts' Law, visual search, and decision in Equations 3, 5, and 6.

SDP can be used for hierarchical or multilevel predictions (ML_i) by summing individual level times and including a 'steering cost' sc_i for traversing from one menu level to the next (e.g., to a menu cascade).

$$ML_i = \sum_{j=1}^{l-1}(T_j + sc_j) + T_l \quad (8)$$

A variant of the SDP model has been used to predict hierarchical navigation in scrolling lists (such as file browsers) [13], but it has not been used with non-linear interfaces such as radial menus, toolbars, or ribbons.

PERFORMANCE PRINCIPLES SUGGESTED BY SDP

Analysis of the SDP model suggests that performance is governed by two main principles: the time to visually scan for an item of interest, and the time to point to it. The model also shows that the relative importance of these two tasks is different for novices and experts. In particular, SDP suggests that novices spend more than half of target selection time visually searching for items, while experts spend most time on the motor aspects of acquisition. Figure 3 shows the predicted percentage of selection time dedicated to pointing (T_p) by novices and experts in single-level traditional menus across various menu sizes, using parameters reported in [14] – the remaining time is dedicated to search and decision (T_{dsi}). Search aids such as Ephemeral Adaption's [16] pre-attentive stimuli will be proportionately more valuable for novice users, while pointing improvements apply throughout the user's transition from novice to expert performance, and are particularly valuable (proportionately) for expert users.

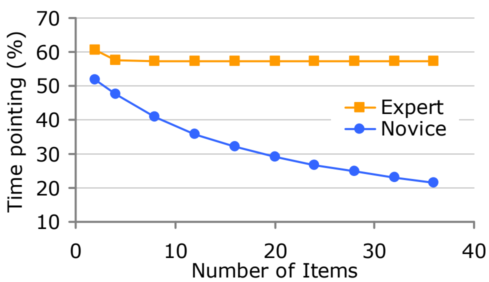

Figure 3. Predictions of the percentage of selection time spent pointing for novices and experts, by menu length.

MENU DESIGNS FOR MINIMIZING POINTING TIME

We identified two types of menus that are likely to perform well given the principles identified above: pie menus, which have been extensively studied (discussed above with Related Work), and Square Menus, a novel type that we designed to improve pointing performance while retaining a rectilinear format.

Square Menus

We designed a new type of menu called Square Menus (Figure 4) to minimize pointing time for experts, while

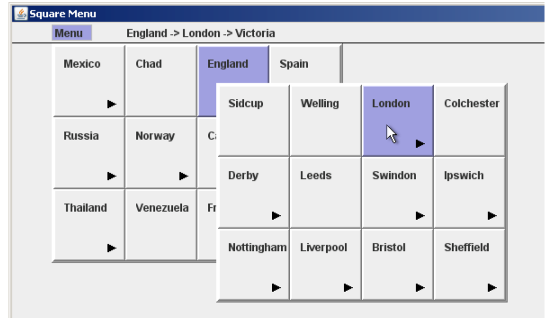

Figure 4. A hierarchical Square Menu.

retaining the basic rectilinear layout of traditional menus. Square Menus arrange menu items in a square (or nearly square) grid, similar in some ways to Microsoft's 'Ribbon' toolbars or the Mac OS X 'file stack.' If a menu level contains n items then the horizontal and vertical grid size is the ceiling of the square root of n, with blank items in the lower right region as necessary. Cascading submenus are accessed by clicking in a parent item. Since submenus can obscure part of the parent menu, they can be explicitly dismissed either by moving out of the submenu, right clicking, or pressing Escape (as with traditional menus).

Square Menus reduce Fitts' Law pointing time compared with traditional linear menus. Figure 5 shows a comparison of the theoretical average pointing times (T_p) with Square Menus and traditional linear menus of various lengths. The predictions use Fitts' Law calibration parameters of a = 0.37 seconds and b = 0.13 seconds/bit from [14]. The figure shows increasing benefits for Square Menus as menu size increases. The predictions do not change with menu item size since Fitts' Law *ID* concerns the ratio of distance to width, which is scale independent.

Pie and Square Menus have theoretical characteristics that fit the performance principles suggested by the SDP model. Our next steps were to build predictive models for the two designs (and for traditional linear menus as a control), and then empirically validate the predictions.

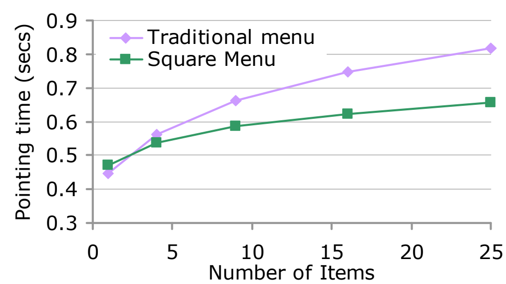

Figure 5. Pointing time for traditional and Square Menus.

EXPERIMENT 1 – MODEL CALIBRATION

The goal of the model is to predict novice and expert performance with various types of hierarchical menus. Calculating predictions requires that intercept and slope parameters be calibrated for Fitts' Law pointing (Equation

3), visual search (Equation 5), and Hick-Hyman decision (Equation 6). Some of these parameters will be the same for different interfaces, as they involve similar or identical interaction mechanics. For example, pointing requirements with traditional menus and Square Menus are similar – the user moves the cursor as normal in either case – so the same Fitts' Law parameters can be used. Sometimes, however, the mechanics are clearly different, demanding calibration: for example, it is unclear how pointing time increases with the number of items in a pie menu. We therefore ran an experiment to calibrate the following model parameters:

Pointing time in pie menus. Given their prominence in research studies, surprisingly little is known about how users interact with pie menus. When pointing to an item in a pie menu, the user only controls cursor direction (in contrast to direction *and* magnitude of movement in traditional pointing). Is selection time therefore related to the number of items, and if so, how (linearly, logarithmically, or other)? How far do users move the cursor as the number of items increases? In calibrating T_p for pie menus, we are primarily interested in the relationship between the number of items and pointing time, but the experiment also allows us to characterise other facets of radial pointing that have not been reported in prior research.

Visual search time for Square Menus and pie menus. There is extensive literature on visual search (see [31] for a review), but it is unclear how sequential visual search times are influenced by target layout (e.g., items arranged in a grid or circle). We therefore calibrated visual search parameters for Square and pie layouts to determine whether they differ from traditional menus. Traditional search times were extracted from [14], which used a very similar procedure to ours.

Apparatus and Participants

The experiment ran on a Windows XP PC with a 22-inch LCD monitor at 1680×1050 resolution and a conventional optical mouse. Twelve volunteer undergraduate students (three female) aged 21 to 39 years (mean 26.4 years, SD 6.0) participated in the experiment. All used computers daily. Participation lasted approximately 35 minutes.

Method

All participants completed four experimental phases: a Fitts' Law calibration phase for traditional pointing, a pointing calibration phase for pie menus, a visual search phase with pie menus, and a visual search phase with Square Menus. The first two phases were always completed in this order, but the third and fourth phases were counterbalanced to mitigate learning and fatigue effects. Participants were instructed to complete trials as quickly and accurately as possible. Identical fonts were used to label items in all conditions.

Fitts' Law calibration. Estimating visual search times requires subtracting pointing time from the total time to find and select an item. We therefore need to know pointing time for each participant, and consequently they completed a rapid Fitts' Law calibration phase involving six *ID* values (from 2.25 to 5.2 bits) constructed from two target sizes (22 and 80 pixels, representing the height of traditional and Square Menu items) and three distances (300, 550, and 800 pixels). Each trial consisted of moving vertically downward from a 'start' button to a square green target. Participants were instructed to click the 'start' button to display the target and not to move the cursor until they had visually acquired the target. Trial time only started as the cursor left the start button (and ended with a click in the target) thus visual search time was excluded. There were a total of 30 trials, consisting of five blocks, with one trial of each *ID* occurring once in a random order in each block.

Pie menu pointing calibration. Participants completed 168 pointing trials with pie menus, using seven menu sizes in a random order (2, 4, 9, 16, 25, 36, and 49 items), each with six blocks of four trials (one in each pie quadrant, or two selections in each half for two-item menus). The pie menu diameter increased with the number of items (101 pixels diameter for two items, to 981 pixels for 49 items). Each trial proceeded as follows. Initially a blank pie menu with the correct number of items (but no labels) was posted to the screen centre, and the cursor was locked by continually warping it to the menu centre. After 600ms, the target was identified by showing a numerical label inside it, and the cursor was unlocked. Highlighting provided feedback of the item under the cursor. Pointing time was measured between the first movement and the selection.

Visual search in pie menus. Participants completed 30 trials (ten blocks, and three items per block, with one item randomly located in each third of the menu) with each of five pie menu sizes in random order: 4, 9, 16, 25 and 36 items. The menus were populated with names of countries, capitals and US states (5-7 characters in length). Each timed trial began by clicking a button, which displayed the menu and showed the target name immediately above it. Selecting an item completed the trial whether correct or not. An incorrect selection added an additional trial, so all participants produced data for 150 correct selections (30 trials × 5 menu sizes). Visual search time is calculated by subtracting pointing time (based on each person's Fitts' Law regression parameters) from the total selection time.

Visual search in Square Menus. This phase used the same procedure as visual search in pie menus. Menu item size was arbitrarily set at 80×80 pixels, and the target cue was shown alongside the button used to post the menu.

Each of the twelve participants completed 498 trials (30 Fitts, 168 pie pointing, 150 pie visual search, and 150 Square Menu visual search), giving 5976 trials in total.

Results

Table 1 summarises the results of linear regression models for pointing, visual search, and Hick-Hyman decision time (values for traditional menus and Hick-Hyman parameters are from [14]). All regression models are extremely good ($R^2 > 0.97$).

	Traditional a b	Square a b	Pie a b
Pointing	0.13+ 0.15×ID	0.13+ 0.15× ID	0.32+ 0.014× n
R^2	0.976	0.976	0.997
Vis. srch	0.3+ 0.08×n	0.32+ 0.12× n	0.40+ 0.12× n
R^2	0.99	0.997	0.999
Hk-Hym.	0.24+ 0.08×H	0.24+ 0.08× H	0.24+ 0.08× H

Table 1. Calibration parameters from Experiment 1. *n* is number of items. *H* is information entropy (see Equation 6).

Characterisation of radial pointing

Figure 6 shows that radial pointing time in pie menus increases *linearly* with the number of items (R^2 = 0.997). This is an interesting result that warrants further investigation because Fitts' Law predicts the logarithmic relationship shown (Figure 6). The Fitts' prediction is calculated using a and b values from [14], and taking W to be the width of the pie slice at a distance D from the pie centre (we use the smaller of the area's two dimensions for W). The ID is then calculated as $\log_2(1/(2\tan(360/2n))+1)$. We are unaware of prior research showing this relationship. Experimental logs also show that the Euclidean cursor movement distance from the menu centre to the selection point increases linearly with the number of items, from 11 pixels with 2 items, to 94 pixels with 49 items (R^2 = 0.98), confirming that users move farther when there are more items. Note, however, that our experimental method increased diameter with n, so movement distance could be a factor of diameter as well as (or instead of) n.

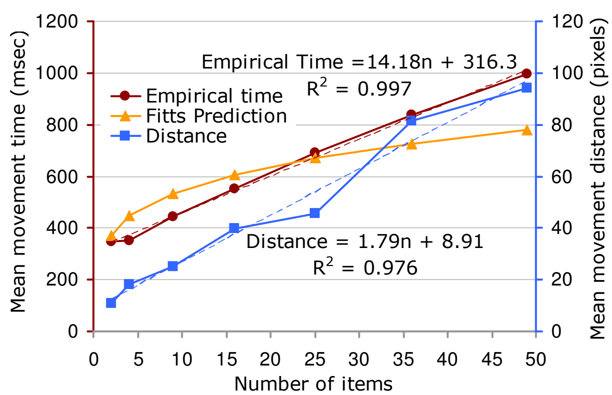

Figure 6. Mean time and distance for pie menu movements.

Observations on visual search

Visual search times increased linearly with the number of items in both Square and pie menus (R^2 = 0.997 and 0.999 respectively). Regression models are similar across the three menu types. Traditional menus offer a slight advantage, which is probably due to their simple and straight visual scan paths, the close proximity of items, and the participants' greater familiarity with the layout.

Surprisingly, there is little evidence that pie menu errors increase with the number of items (R^2 = 0.14). Furthermore, the total number of errors is identical with Square and Pie menus (1.6% error rate). Unlike gestural marking menus, which have been shown to suffer high error rates with more than eight items [27], pie menu users can maintain accurate selections by attending to feedback and moving further to give a larger target area.

EXPERIMENT TWO – TESTING THE MODEL

The second experiment tests the accuracy of the model's predictions for hierarchical menu selection performance with the three interfaces, as users gain expertise.

Method

Apparatus and Participants. The apparatus was identical to experiment one. Twenty-four undergraduate students (six female) aged 19 to 49 years (mean 29.1 years, SD 8.3) took part in the experiment. All used computers daily. None were participants in experiment one.

Traditional menu items were 120×22 pixels; square menu items were 80×80 pixels; 12- and 16-item pie menus were 301 and 411 pixels in diameter.

Task Data. The menu hierarchies used in the experiment consisted of sixteen countries at the first level, up to sixteen cities at the second, and up to twelve suburbs at the third. Exactly the same structure was used with all three interfaces, but the structures were populated with different data. Identical structures are rendered entirely differently in the three interfaces, making it unlikely that participants would notice the reuse or profit from it.

Procedure and Design. All participants completed ten familiarisation trials, then eight blocks of experimental trials with all three menu systems (order balanced using a Latin square). Each block reused the same six targets in a random order, allowing participants to gain experience. Two targets occurred in a second level menu (e.g., "Chile⇨Santiago"), and four occurred in a third level sub-submenu (e.g., "France⇨Paris⇨Olympia"). The top level menu contained sixteen countries, twelve of which had associated submenus. Second level targets always occurred in submenus containing twelve items, and third level targets always occurred by navigating through a submenu of twelve to a sub-submenu of sixteen. The six targets occurred in the same structural position for all participants in all conditions, as follows: (6, 4), (9, 10), (3, 3, 15), (7, 6, 3), (11, 9, 8), and (14, 11, 11), where the tuples represent locations in the first, second and third levels. Entirely different sets of countries and places were used with each interface. A click in a 'Menu' button posted the first menu level, displayed the target path and started timing. A mouse button release inside a parent item (or after a dwell time of 333ms in the traditional menu) posted submenus. Timing ended when the mouse button was released inside a non-parent item. An incorrect selection added an additional trial.

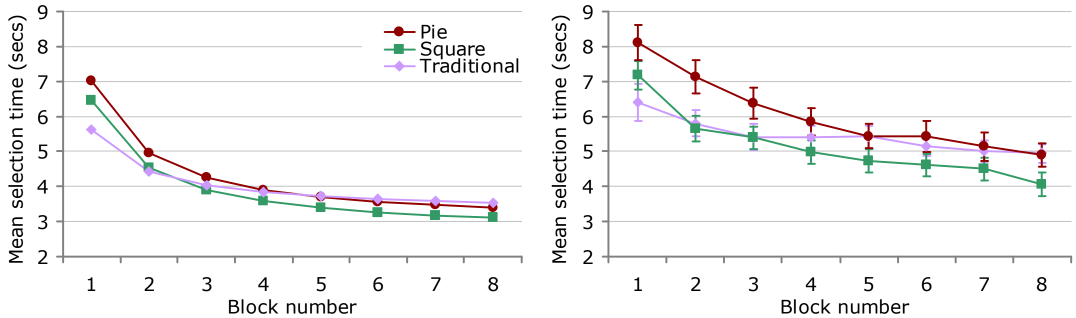

Figure 7. Predicted (left) and empirical (right) results for the three interfaces. Error bars show ± 2 SE.

We gathered data from 3456 correct trials: 24 participants × 3 interfaces × 8 blocks × 6 targets. All data from one participant was discarded as it accounted for 29 of 52 outlier trials (more than 3 SD from the mean).

Predicting Performance with the Calibrated Model

The calibrated parameters shown in Table 1 were used in a simple spreadsheet to predict performance for these experimental tasks. Visual search parameters for linear menus were extracted from [14], as were Hick-Hyman decision time parameters. We treat each menu level as a separate Hick-Hyman decision: for example, first deciding about the location of "Canada", then "Toronto", and then "Bolton". The probability associated with each decision is $^1/_{16}$ (the reciprocal of the number of selections in each block). It is possible, however, that users actually retrieve the full hierarchic structure from memory in one process, causing us to overestimate decision times. We revisit this issue in the discussion. Finally, the steering cost for hierarchical items in traditional menus is calculated by adding the horizontal traversal distance between the centre of the parent item and the horizontal centre of its submenu to the total movement amplitude for the cascading item. The predictions across eight trial blocks are shown in Figure 7.

Results

We divide our results into five sections: the comparison between predicted and empirical results (to test the model); comparison of the menu designs themselves; analysis of cumulative times through the hierarchy; data concerning the low predictions; and participant comments.

Results 1: Testing the model

Figure 7 (left) shows the model's predictions of mean target selection time for the three interfaces, and the empirical results (right). Linear regression of predicted versus empirical data gives R^2 values of 0.96, 0.93 and 0.92 for the Square, traditional and pie menus respectively.

There are two aspects of the predictions that we evaluate below – the relative and the absolute accuracy.

First, the model performs well in predicting the relative performance of the three menu designs. In particular, it correctly predicts four characteristics of the data:

- *The order of the designs for novice usage*. The model correctly predicted the order of the menu designs in block one – that pie menus would be slowest, traditional menus fastest, and Square Menus in between.
- *The order of the designs across blocks*. The model also correctly predicted the order of the designs once participants were familiar with the items – Square fastest, and pie and traditional menus very close together, but with pie slightly faster.
- *The crossover with increasing expertise*. The model correctly showed that there would be a crossover between traditional and Square Menus at about block two, and that performance of pie and traditional menus would meet at about block five.
- *The overall shape of the curve*. The model predicted a power-law learning effect for all three menu designs, and this is confirmed (although with more noise than predicted) in the empirical data.

Second, in terms of absolute accuracy, Figure 7 shows that the model underestimates completion times by approximately one second. We discuss possible reasons for this underestimation in Results 4, but we note here that this absolute error is something that can be improved in future refinements to the model.

Results 2: Empirical Comparison of the Designs

We are also interested in the empirical comparison of the three menu designs – that is, whether any of the menus is significantly faster than the others. We carried out a 3×8 ANOVA to test the effects of *menu design* and *block* on selection time. The test showed a significant main effect of *menu design* ($F_{2,44} = 17.45$, $p < 0.001$), with Square fastest at 5.14 seconds, then traditional (5.45 seconds), and pie (6.04 seconds).

As anticipated, there was a significant effect of *block* ($F_{7,154} = 63.49$, $p < 0.001$). There was also a significant *interface×block* interaction ($F_{14,308} = 5.92$, $p < 0.001$), caused by the cross-over effect of interface performance with experience (Figure 7).

To investigate the interaction, we carried out a second ANOVA using only the final three blocks (i.e., those where participants were most experienced). We again found a significant main effect of menu design ($F_{2,44} = 10.18$,

$p < 0.001$), with means of 4.39s (Square), 5.04s (traditional), and 5.16s (pie). A post-hoc Bonferroni test showed that Square Menus were significantly faster than either traditional or pie menus ($p < 0.01$); there was no significant difference between traditional and pie ($p = 0.1$).

Results 3: Cumulative times at hierarchical levels

Figure 8 shows predicted and empirical cumulative times through hierarchical levels with the three interfaces. Time measures at each level are taken when either a non-parent item at that level is selected (with a mouse button release) or when the submenu is posted by a mouse button press in its parent item (or for the traditional menu, after the dwell timeout). It shows that the empirical data is higher than predicted for the first level, but relatively consistent with predictions for the remaining levels. The mean prediction errors at the first, second, and third levels are 45, 30 and 19% respectively.

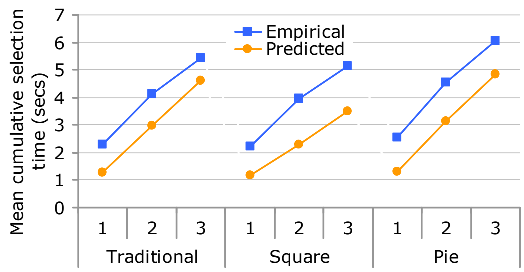

Figure 8. Predicted and empirical cumulative times at the 1st, 2nd, and 3rd hierarchical levels with the three interfaces.

Results 4: Underestimation of absolute selection times

Although the model successfully predicts important performance trends (such as the performance crossover with experience), it underestimates task times (at worst by 34% for pie menus in block 6). While this is an obvious weakness, we believe it can be readily explained.

Figure 8 shows that the discrepancy arises primarily at the first hierarchical level. This is best explained by differences in reading and interpreting the target cue during calibration and experiment two. Calibration tasks involved reading a one word cue, finding that word/item, and clicking on it. The cues used in experiment two, however, were much longer, involving three strings (such as "South Africa⇨Durban⇨Sunnyhills"). Reading, processing, and memorizing these multi-part hierarchical cues is much harder than in the calibration experiment, and results in slower than predicted performance. Importantly for the success of the model, this difference is an artifact of our experimental method, and not an artifact of the underlying interaction that we wish to understand.

A secondary cause of the discrepancy for the traditional menu arises from participants waiting for dwell timeouts to expire rather than by explicitly clicking the parent. Approximately 79% of trials with the traditional menu included waiting for timeouts to expire. Nine participants always waited for the timeout to expire.

Results: Participant comments

All participants were asked to state which menu system they found fastest, easiest to move through, and easiest to remember item locations. Eleven participants stated that Square Menus were fastest, eight pie menus, and four traditional. Thirteen participants stated that pie menus were easiest to move through (eight for Square and two for traditional), with many praising their "quick and short" selection gestures. Opinions were split on which best facilitated location memory, with fourteen for Square, seven for traditional, and two for pie.

DISCUSSION AND FUTURE WORK

Results show that the model successfully predicts subtle performance trends for the three menu types. Most notably, it predicts the reversal of the performance order between initial use and use after familiarisation, and it attributes the change to the relative efficacy of visual search and pointing mechanics with the three interfaces. The model was less successful at predicting absolute task times; however, we believe this effect is primarily due to differences in how tasks were experimentally cued, as discussed in the results section, rather than being a failure in the model itself.

Square Menus: Design and Refinements

Square Menus performed well in the experiment, and participants were enthusiastic about them. Several participants preferred and performed best with pie menus once they had gained some expertise, but Square Menus offer several pragmatic advantages as well – including their applicability at any screen location (whereas pie menus are awkward to render and use at window edges), their simple layout, and their support for broad structures (see below).

As our focus is on modelling, our current Square Menu design is rudimentary, but many visual and interactive embellishments are possible. These include most of the traditional menu enhancements discussed in related work, such as enhanced visual identification with Ephemeral Adaptation [16] or improved target acquisition with Bubbling menus [36]. In terms of modelling, these enhancements would lower the a and b parameters associated with visual search and pointing. We are also keen to experiment with transparency effects to help users maintain awareness of items occluded by hierarchical menus [21].

Pie menus: Questions about Performance

One question raised by the empirical results is why pie menus did not outperform traditional, even though pie menus had good theoretical characteristics as described earlier. We believe that a difference would eventually appear between these two designs, but that our study did not provide enough blocks to reveal it. As can be seen in the predicted performance (Figure 7), the separation between pie and traditional menus only begins to occur at about block eight, which was the endpoint of our study. In addition, we note that there are likely to be different training curves for different interfaces, and development of expertise may simply take longer with pie menus. This has practical importance in that it will take longer for users to 'pay back' the initial costs incurred with a pie design.

Refinements and Extensions to the Model

The study suggests two extensions to SDP: modelling of Hick-Hyman decision for composite hierarchical actions, and modelling of spatially unstable designs.

First, the model currently uses separate Hick-Hyman decision time values at each level of the hierarchy, but it is likely that experienced users make a single composite decision about spatial selection actions. For example, a frequent pie menu selection might involve left, down, and right gestures across three levels, which with expertise becomes a single decision to produce a '↳' shape. We will return to this transition in future work.

Second, we will investigate additional modelling of spatial stability and instability in menu designs. Square Menus maintain spatial stability to help users learn item locations. Although this may seem an obvious design decision, there is an important tradeoff to consider. Context sensitive menus, which alter the set of commands displayed dependent on the user's activity, can assist novice visual search because all irrelevant controls are removed (the Microsoft Ribbon uses this approach, varying toolbar items based on the selected object; Square Menus instead fade and disable irrelevant items). Modelling performance with context-sensitive behaviour is complex, demanding much richer understanding of user tasks and system state than can be achieved with our model's simple formulae governing search, decision, and pointing. However, the model helps identify the main potential benefits and costs, and extensions should be able to characterise this tradeoff: context sensitivity will sometimes aid novice performance, but the lack of spatial stability may make it more difficult to develop expertise with item locations.

Use of the Model in Design Scenarios

The experiment shows that the model correctly predicted a ~12% benefit for traditional over Square Menus when novice, and a ~15% benefit in the opposite direction after familiarisation. Relatively small differences like these may be important for designers in particular contexts, such as when determining the optimal interface for dedicated use in a call centre. Frequently, however, designers will face coarser questions that cannot be easily answered without empiricism, and then the empiricism may mislead them. For example, a designer who needs to provide access to 36 items (of six categories) might be considering a single Square Menu and a hierarchical pie menu. Figure 9 shows predictions for equally probable targets in these structures; initially pie menus are 40% faster than Square Menus, but with experience, performance quickly crosses over until Square Menus are ~35% faster. These predictions were calculated in minutes using a standard spreadsheet.

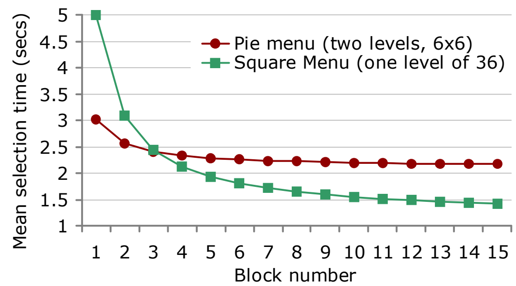

Figure 9. Predictions for 36 items in a flat Square Menu versus a hierarchical pie menu.

CONCLUSIONS

Interface designers make important choices about the widgets and structures used to access commands. While empirical evaluation is a critical tool in guiding their decisions, it is time consuming, and therefore limits the range of alternatives, layouts, and levels of user expertise that can be measured.

We have applied earlier modelling work on single-level linear menus to hierarchies of non-linear designs such as pie menus. The model motivated the design of Square Menus, which arrange items in a grid, reducing Fitts' Law pointing time. Calibrating the model also revealed new results that radial pointing time and movement distance increase linearly with the number of pie menu items. An experiment showed that the model accurately predicted important performance trends, such as a the superiority of Square Menus for experts, and the reversal of relative interface performance with expertise – novice users performed best with traditional menus and worst with pies; users with more expertise were worst with traditional. The strong performance of Square Menus is encouraging because they do not encounter some of the layout and window-edge problems of pie menus (although future studies should investigate some of the additional strengths of radial menus in this context, such as gestural shortcuts).

Overall, our experiences suggest that the model's simple equations can help designers quickly consider the performance impact of many alternative designs and experience levels without need for time-consuming implementation and experimentation.

REFERENCES

1. Accot, J. and Zhai, S. Beyond Fitts' Law: Models for trajectory-based HCI tasks. In *Proc. CHI 1997*, ACM Press (1997), 295-302.
2. Ahlström, D. Modeling and improving selection in cascading pull-down menus using Fitts' law, the Steering law and force fields. In *Proc. CHI 2005*, ACM Press (2005), 61-70.
3. Ahlström, D., Alexandrowicz, R. and Hitz, M. Improving menu interaction: A comparison of standard, force enhanced and jumping menus. In *Proc. CHI 2006*, ACM Press (2006), 1067-1076.

4. Bailly, G., Lecolinet, E. and Nigay, L. Flower Menus: A new type of marking menus with large menu breadth, within groups and efficient mode memorization. In *Proc. AVI 2008*, ACM Press (2008), 15-22.
5. Bailly, G., Lecolinet, E. and Nigay, L. Wave Menus: Improving the novice mode of hierarchical marking menus. In *Proc. INTERACT 2007*, 475-488.
6. Balakrishnan, R. "Beating" Fitts' Law: Virtual enhancements for pointing facilitation. *Int. Journal of Human Computer Studies 61*, 6 (2004), 857-874.
7. Byrne, M.D. ACT-R/PM and menu selection: Applying a cognitive architecture to HCI. *Int. Journal of Human Computer Studies 55,* 1 (2001), 41-84.
8. Byrne, M.D., Anderson, J.R., Douglass, S. and Matessa, M. Eye tracking the visual search of click-down menus. In *Proc. CHI 1999*, 402-409.
9. Callahan, J., Hopkins, D., Weiser, M. and Shneiderman, B. An empirical comparison of pie vs. linear menus. In *Proc. CHI 1988*, 95-100.
10. Card, S.K. Visual search of computer command menus. In Bouma, H. and Bouwhuis, D.G. (Eds.), *Attention and Performance, X: Control of Language Processes*, Erlbaum, London, 1984, 97-108.
11. Card, S.K., Moran, T.P. and Newell, A. *The Psychology of Human-Computer Interaction*. Erlbaum Associates, Hillsdale, NJ, 1983.
12. Cockburn, A. and Gin, A. Faster cascading menu selections with enlarged activation areas. In *Proc. GI 2006*, Canadian Inf. Processing Society (2006), 65-71.
13. Cockburn, A. and Gutwin, C. A predictive model of human performance with scrolling and hierarchical lists. *HCI 24*, 3 (2008), 273-314.
14. Cockburn, A., Gutwin, C. and Greenberg, S. A predictive model of menu performance. In *Proc. CHI 2007*, ACM Press (2007), 627-636.
15. Findlater, L. and McGrenere, J. A comparison of static, adaptive, and adaptable menus. In *Proc. CHI 2004*, ACM Press (2004), 89-96.
16. Findlater, L., Moffatt, K., McGrenere, J. and Dawson, J. Ephemeral adaptation: The use of gradual onset to improve menu selection performance. In *Proc. CHI 2009*, ACM Press (2009), 1655-1664.
17. Fitts, P.M. The information capacity of the human motor system in controlling the amplitude of movement. *J. Exp. Psychology 47* (1954), 381-391.
18. Greenberg, S. and Witten, I.H. Supporting command reuse: Empirical foundations and principles. *Int. J. of Man-Machine Studies 39,* 3 (1993), 353-390.
19. Grossman, T., Dragicevic, P. and Balakrishnan, R. Strategies for accelerating on-line learning of hotkeys. In *Proc. CHI 2007*, ACM Press (2007), 1591-1600.
20. Hanson, S.J., Kraut, R.E. and Farber, J.M. Interface design and multivariate analysis of UNIX command use. *ACM TOIS 2*, 1 (1984), 42-57.
21. Harrison, B.L. and Vicente, K.J. An experimental evaluation of transparent menu usage. In *Proc. CHI 1996*, ACM Press (1996), 391-398.
22. Hick, W.E. On the rate of gain of information. *The Quarterly Journal of Exp. Psych. 4,* 1 (1952), 11-26.
23. Hornof, A.J. and Kieras, D.E. Cognitive modeling reveals menu search is both random and systematic. In *Proc. CHI 1997*, ACM Press (1997), 107-114.
24. Hyman, R. Stimulus information as a determinant of reaction time. *J. Exp. Psych. 45,* 3 (1953), 188-196.
25. Kobayashi, M. and Igarashi, T. Considering the direction of cursor movement for efficient traversal of cascading menus. In *Proc. UIST 2003*, ACM Press (2003), 91-94.
26. Kurtenbach, G. and Buxton, B. The limits of expert performance using hierarchic marking menus. In *Proc. InterCHI 1993*, ACM Press (1993), 482-487.
27. Kurtenbach, G.P., Sellen, A.J. and Buxton, W.A.S. An empirical evaluation of some articulatory and cognitive aspects of marking menus. *HCI 8*, 1 (1993), 1-23.
28. Lane, D.M., Napier, H.A., Batsell, R.R. and Naman, J.L. Predicting the skilled use of hierarchical menus with the Keystroke-Level model. *HCI 8*, 2, (1993), 185-192.
29. Lee, E. and McGregor, J. Minimizing user search time in menu retrieval systems. *Human Factors 27*, 2 (1985), 157-162.
30. Newell, A. and Rosenbloom, P.S. Mechanisms of skill acquisition and the law of practice. In Anderson, J.R. (Ed.) *Cognitive Skills and their Acquisition*, Erlbaum, Hillsdale, NJ, 1981, 1-55.
31. Oakley, I., Brewster, S. and Gray, P. Solving multi-target haptic problems in menu interaction. *Ext. Abstracts CHI 2001*, ACM Press (2001), 357-358.
32. Sears, A. and Shneiderman, B. Split menus: effectively using selection frequency to organize menus. *ACM TOCHI 1*, 1 (1994), 27-51.
33. St. Amant, R., Horton, T.E. and Ritter, F.E. Model-based evaluation of expert cell phone menu interaction. *ACM TOCHI 14*, 1 (2007), 1-24.
34. Tanvir, E., Cullen, J., Irani, P. and Cockburn, A. AAMU: Adaptive activation area menus for improving selection in cascading pull-down menus. In *Proc. CHI 2008*, ACM Press (2008), 1381-1384.
35. Thimbleby, H. *Press On: Principles of interaction programming*. MIT Press, Cambridge, MA, 2007.
36. Tsandilas, T. and schraefel, m.c. Bubbling menus: A selective mechanism for accessing hierarchical drop-down menus. In *Proc. CHI 2007*, 1195-1204.
37. Wolfe, J. What can 1 million trials tell us about visual search? *Psychological Science 9*, 1 (1998), 33-39.
38. Zhao, S., Agrawala, M. and Hinckley, K. Zone and polygon menus: Using relative position to increase the breadth of multi-stroke marking menus. In *Proc. CHI 2006*, ACM Press (2006), 1077-1086.
39. Zhao, S., Dragicevic, P., Chignell, M., Balakrishnan, R. and Baudisch, P. earPod: Eyes-free menu selection using touch input and reactive audio feedback. In *Proc. CHI 2007*, ACM Press (2007), 1395-1404.