Journal of Computer Science 5 (12): 937-947, 2009
ISSN 1549-3636

Does the Size of Personalized Menus Affect User Performance?

Khalid Al-Omar and Dimitrios I. Rigas
Department of Computing, University of Bradford, Bradford, UK

Abstract: Problem statement: To date, researchers have personalized graphical user interfaces for individual users to reduce visual search time. Little research, however, has been directed at understanding the factors which cause approaches to personalization to have positive effects at one time and negative ones at others. **Approach:** The study reported here investigated empirically the effects of content size on 5 different personalized menu types: Adaptable, adaptive split, adaptive/adaptable highlighted, adaptive/adaptable minimized and mixed-initiative menus. More specifically, it compared the usability of these five types with regard to task accomplishment time and frequency of error-occurrence. In order to carry out this comparative investigation, we conducted two independent experiments, on small menus (17 items) and large ones (29 items) respectively. These were tested dependently using 30 subjects each. **Results:** Results showed that the adaptable type was surprisingly the most efficient overall of the small menus and the least efficient of the large ones. **Conclusion:** Conversely, the minimized type was the slowest of the small menus and the fastest of the large ones. Finally, errors were reduced in adaptable and minimized small menus by 50 and 62% respectively, whilst being increased in the large adaptable one.

Key words: Adaptive, adaptable, mixed-initiative, menus, performance, interactive systems, usability

INTRODUCTION

Interface visual complexity is increasing rapidly with each new release of software, as the numbers of icons, menus and toolbars grow and sizes of the interface formats decrease (e.g., mobile phones, PDAs). This visual complexity has become recognized as a phenomenon which some researchers call creeping featurism[1] and others bloatware[2,3]; it creates conditions where usability problems can arise[3] and where user performance and satisfaction are affected negatively. In response, researchers have sought methods to organize and control such interfaces. By way of illustration, McGrenere[4] has suggested multiple interfaces as a solution to software complexity. Other studies have focused on organizing interfaces by using different techniques. For example, some menu designs have used sorting techniques such as alphabetical order and categorical color-coding[5], while others have focused on visualizing techniques. For example, circular menus have been developed so that all menu items are equally distant[5]. Another example is the adaptive split menu for font selection in Microsoft Word 2000. Alternatively, a number of researchers have suggested personalizing interfaces to individual users, since they have different needs, abilities and usage[6].

There are three approaches to personalization. Adaptive interfaces dynamically change the interface layout and content to each user's needs, while adaptable interfaces provide customization techniques which permit users to adjust their layout and content to suit their needs. Mixed-initiative interfaces combine these two approaches to provide what best suits the user[7]. These approaches differ in their control of personalization: Adaptive approaches are system controlled, adaptable approaches are user controlled and mixed initiative approaches are both system controlled and user controlled at the same time[6]. In addition, there are differences in the techniques they tend to use. For example, adaptive interfaces have tended to use graphical or spatial techniques, or a combination of both, to reduce visual search time[8]. Graphical techniques recognize items and change them graphically, whereas spatial techniques recognize such items and move or copy them for easier access. Adaptive split menus, for example, move the most frequently or recently used items to the top of the menu[7]. A recent development is the introduction of ephemeral menus, which reduce search time by presenting predicted items immediately, while remaining items gradually fade in[8]. For their part, adaptable interfaces have tended to use coarse or fine graining, or a combination of both, to reduce visual

Corresponding Author: Khalid Al-Omar, Department of Computing, University of Bradford, Bradford, UK

complexity[6]. Coarse-grained menu control allows users to move items to the top or bottom, whereas fine-grained control lets them move items to specific positions in the list. For example, such techniques are utilized in adaptable split menus to allow users to move items to the top or bottom partition[6].

To date, far too little attention has been paid to comparing the adaptable, adaptive and mixed-initiative approaches, although there has been some debate in the field of human-computer interaction as to which of these approaches is best[9]. One side argues that users should be able to manage their tools easily, while the other believes that they need the right adaptive algorithms[6]. Despite this debate[8], points out that research to date has tended to focus on proving one side of the argument, rather than understanding the factors making some of approaches successful in one context and less so in another. The few exceptions include a study examining the impact of screen size on performance, satisfaction and awareness in the adaptive approach[10], which found that screen size affected users' behavior. Subjects took advantage of the adaptive predictions more often in small screen conditions compared to a large screen. It was found that small screens were slower than large ones, because scrolling took time but was unnecessary with a large screen. In addition, adaptive accuracy had a larger positive effect on performance and satisfaction in small screens than in large ones. The authors of another empirical study[11] compared their own results with those of other relevant studies[6,12,13] and suggested a number of vital factors that could affect the success of an adaptive interface. These included spatial stability, accuracy and frequency of adaptation, frequency of interaction with the interface and the complexity of the tasks and of the interface itself.

The primary contribution of this study is to provide empirical comparison concerning the effect of menu size on the performance of personalized menus. More specifically, it measures the effect of small[25] Vs large[24] menu size on the performance of adaptive (split menu and both highlighted and minimized menus in block (1), adaptable (adaptable and both highlighted and minimized menus in block (2) and mixed-initiative menus.

Previous work: Many researchers have sought to reduce menu selection time by making recently and frequently selected items easier to choose. An examination of the current research on personalization reveals contradictory findings, including in direct comparisons of adaptive and static approaches. For example, a static interface was compared to three adaptive alternatives as follows: (1) split interface, where important functions were copied into an extra toolbar; (2) moving interface, where important functions were moved into a toolbar and (3) visual popout interface, where important functions were moved and made visually prominent. Two experiments were conducted. The first had 26 participants and investigated the impact of the different interfaces under two adaptive algorithms (frequency Vs recency based). The results showed little difference between the interfaces for the cognitively more complex task, while on the less complex one, the split and moving adaptive interfaces were faster than the static interface. Furthermore, in terms of satisfaction, perceived benefit and perceived cost, the split and moving adaptive interfaces were found most beneficial and least costly and they were preferred in the more complex task. The visual popout interfaces were found distracting. In the less complex task, there was less support for the adaptive interfaces. The second experiment was conducted with 8 participants and compared adaptation accuracy (70 Vs 30%). The results showed that user performance worsened as the adaptive algorithm's accuracy decreased. Another between-subjects study with 40 participants examined an adaptive approach to command line usage[14]. It compared (1) a command-line interface, (2) a menu-based interface, (3) a hybrid interface, where participants had access to both the menus and the command line and (4) an adaptive interface, where the system moved users from the menus to the command line. It was found that the adaptive interface was significantly faster than the non-adaptive, hybrid approach. Another study compared the performance of adaptive and static menus[12]. In a controlled experiment, 26 subjects were asked to search for names in a telephone directory accessible through a hierarchy of menus and this was tested against a static system. Subjects performed faster with the adaptive system, which 69% of them preferred. In addition, results showed that the adaptive system reduced the search paths for repeated names, reduced time per selection by 35% and reduced errors per menu by 40%. Trevellyan and Browne[15] replicated this experiment with a larger number of trials because they believed that subjects would eventually become familiar with the static menu and memorize the required sequence of key-presses. They found that the adaptive system was effective and that after using it for a long period of time users did begin to perform better with the static interface. Another study compared an adaptive menu with a static one. In a controlled experiment, sixty-three subjects were requested randomly to complete 24 tasks using both menus. The results showed that the static menu was faster than the adaptive menu on the first group of tasks, while there was no difference in the second group of tasks between the static and dynamic menus, because subjects in both groups were able to increase their performance significantly. Eighty-one percent of the subjects preferred the static to the adaptive menu[16]. More recently, a study examined a new adaptive technique called ephemeral adaptation. Ephemeral menus present predicted items immediately, while remaining items gradually fade in[8]. These new techniques were examined with static and highlighted adaptive menus. The results showed that ephemeral menus were faster and preferred over the static control condition when adaptive accuracy was high and no slower when adaptive accuracy was low. In addition, ephemeral menus were faster than highlighted adaptive menus, while both were preferred to static menus.

Direct comparisons of adaptive and adaptable approaches have also had conflicting results. For example, a 6 week field study with 20 participants evaluated two interfaces combined with adaptive menus in the commercial word processor MSWord 2000. These were a personalized interface containing desired features only and a default interface with all the features. During the first four weeks of the study participants used the adaptable interface, then used the adaptive interface for the remaining time. It was found that 65% of them preferred the adaptable interface, 15% favored the adaptive interface and the remaining 20% chose the MSWord 2000 interface. However, according to[17], there were two potentially confusing variables. First, MSWord 2000 and the proposed interfaces had very different designs, which may have differed in their usability. Second, all participants completed the adaptive condition after the adaptable condition. In another study, McGrenere *et al*.[3] carried out a controlled laboratory experiment with 27 participants to compare the efficiency of three of the Sears and Schneiderman[13] split menus. The first of these was a static split menu, the second an adaptable split menu where the top half was adaptable by the user and the third an adaptive split menu, where the system would dynamically assign the top half based on frequency and recency of selection. The experiments found no interactive effect between order and menu. On the other hand, the comparison was complicated, according to[17], because performance depended on menu order and subjects were exposed to the three conditions, although when they were not presented with the adaptable interface they were significantly faster with the adaptive or static ones. The findings were that split static menus were significantly faster than adaptive menus. The adaptable menu was faster than the adaptive menu when participants were guided by example, because they were able to understand the value of customization. In addition, results showed that in these circumstances there was no significant difference between the adaptable and static menus. Nevertheless, 55% of subjects preferred the adaptable menu, 30% the adaptive and 15% the static. In another laboratory experiment with 18 participants, Jameson and Schwarzkopf directly compared automatic recommendations controlled updating of suggestions and a condition where no recommendations were available. The comparison was concerned with content rather than the graphical user interface. In the automatic recommendation (i.e., adaptive) system, the updating was performed automatically by the system, while in the (adaptable) system using controlled updating of recommendations, it was done by users and in the third (static) system, no recommendations were provided to users and the system did not change during usage. Jameson and Schwarzkopf found no differences in performance score among the three conditions.

Most studies in the field of personalization have been limited to the differences and similarities among the static, adaptive and adaptable approaches. Consequently, there has been a small amount of research into mixed-initiative interfaces, including a study which compared an adaptive bar (mixed-initiative system) with the built-in toolbar present in MSWord (adaptable system)[18]. This found that the mixed-initiative system significantly improved performance on one of two experimental tasks. In another study, Burnt *et al*.[19] designed and implemented the Mixed-Initiative Customization Assistance (MICA) system, which provided subjects with the ability to customize their interfaces according to their needs, while also providing them with system-controlled adaptive support. They found that users preferred mixed-initiative support and that the MICA system's recommendations improved time on tasks and decreased customization time. Another study compared directly the static, adaptive, adaptable and mixed-initiative approaches to determine which was best in terms of efficiency[20], effectiveness[21] and user satisfaction[22] in e-commerce. It was found that subjects were faster in the mixed-initiative approach, followed by the adaptable and adaptive conditions; they were slowest in the static condition. Furthermore, subjects in the mixed-initiative made fewer errors than those in the other conditions. The highest number of errors was made in the static approach.

Conditions: Figure 1 shows the layout in the experiment and Fig. 2 shows the layout of the menu conditions.

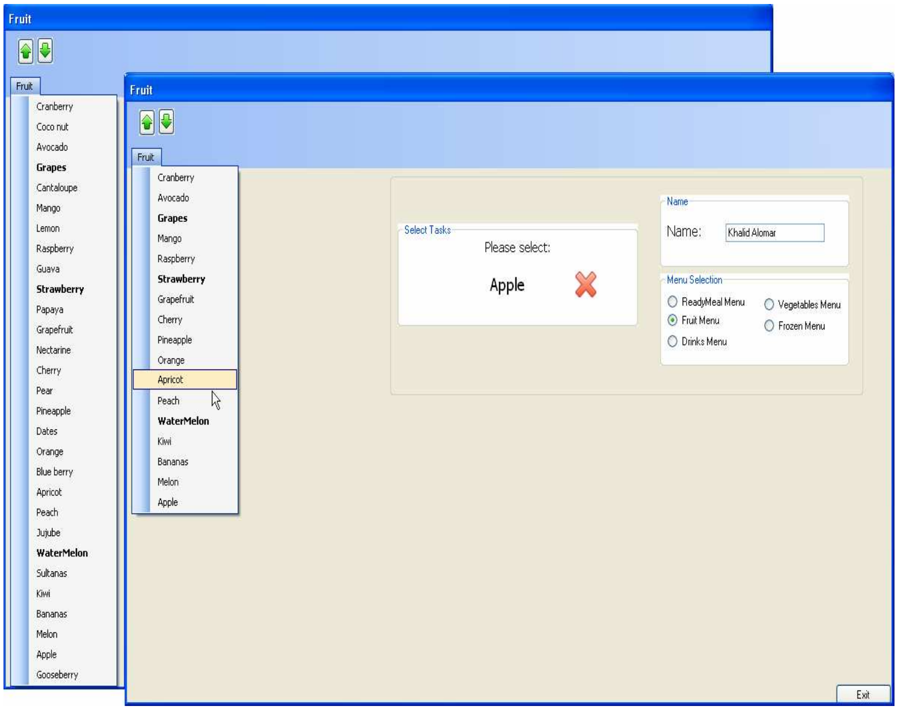

Fig.1: Screen layout in the experiment

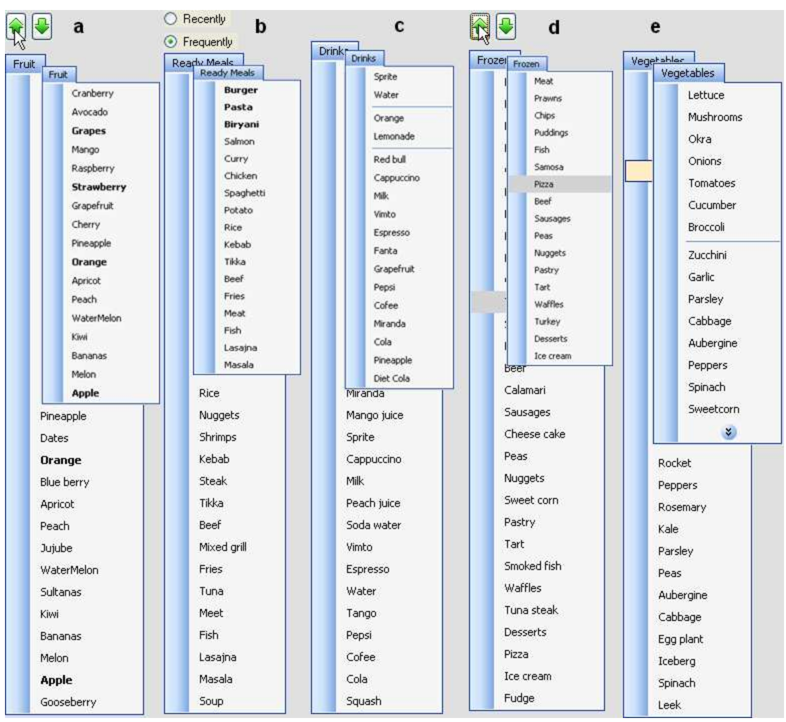

Fig. 2: (a) Adaptive highlight menu (b) mixed-initiative menu (c) adaptive split menu (d) adaptable menu and (e) adaptive/adaptable menu (minimized and hide unused menu items)

Menu size: The large menu was a full length menu displayed on a large screen. The large menu contained 29 items, of which 14 were included in the experiment tasks. The small menu contained 17 items, of which 14 were included in the experiment tasks. The small menu was the size of many menus that are commonly used and was the minimum length that would allow the same number of items (14) to be included in the tasks as for the large one. In addition, it was approximately half the length of the larger menu. If there were different results between this menu size and large menus, it was expected that smaller ones would give better results.

Menu type: Five different menu conditions were tested in each of two experiments (on small and large menus): Adaptable, adaptive split, adaptive/adaptable highlighted, adaptive/adaptable minimized and mixed-initiative menus. The aim was to understand subjects' behavior under the adaptive, adaptable and mixed-initiative conditions and how it varied with menu size; in other words, to explore the impact of size on these five menu conditions. Within the adaptive approach, the chosen techniques were split, highlighted and minimized menus, because their use is commonly reported in the literature with successful results.

In the adaptable condition, subjects could modify the order of items by moving them up or down. This occurred after the first session of the experiment (50 selections). The adaptive split menu was divided by two horizontal lines into three sections. The top section comprised the two most frequently selected items, the second section the two most recently selected items and the bottom section the others. The menu software counted how many times each item had been used in the 50 most recent selections and updated the list after each selection. In the adaptive/adaptable highlighted menu, the most frequently selected items were boldfaced, while the others were not. After the first 50-selection session of the experiment, subjects could modify the order of items by moving them up or down. In the adaptive/adaptable minimized menu, the software counted how many times each item had been used, moving frequently selected items to the top of the list and separating them from other items by a horizontal line. The top section was extendable and kept the most frequently selected items separate from the bottom section. When the user wanted to modify and customize the menu, it would be divided by two horizontal lines into three sections: The top one held the two most frequently selected items, the second comprised the two most recently selected items, while the bottom section contained the others and was hidden. Users could view the hidden items by clicking a small arrow at the end of the menu. In the mixed-initiative menu, the technique was to display the recently or frequently used items to subjects at the appropriate time. The recently selected items were displayed at the top of the menu when this feature was selected by clicking on a button labeled 'Recently', while the frequently selected items were displayed when the 'Frequently' button was selected. Both techniques boldfaced the recently or frequently selected items and moved them to the top. Subjects were able to choose only one technique at a time but could switch from one to another at any time during the experiment. It was the subjects' responsibility to choose the appropriate technique. Figure 1 illustrates the 5 menu types tested in experiments 1 and 2.

Design: A two-factor mixed design was utilized: Menu size (small Vs large) was tested between subjects, while menu type (adaptable, adaptive split, adaptive/adaptable highlighted, adaptive/adaptable minimized and mixed-initiative) was compared within subjects.

Subjects: A total of 60 graduate or undergraduate students voluntarily participated, 30 each on small and large menu designs. These were split 16/14 and 19/11 respectively between males and females. The ages of subjects in both experiments ranged from 18-44, while their average computer usage exceeded 12 h $week^{-1}$. In both experiments, each subject was randomly assigned to one of 5 groups of 6 subjects, each of which followed the 5 experimental menu conditions in a different order. Subjects were given one recorded tutorial according to the experiment they participated in.

Apparatus: In both experiments an application program was developed using Microsoft Visual Basic.Net. Personal computers with Pentium IV 1.5 GHz processors and 17 inch monitors were used in the experiment.

Menu labels: In the small menu experiment, 85 different nouns from five label categories (17 nouns in each category) were used as labels of the menu items, while for the large menus, there were 145 different nouns from the five label categories (29 in each category). The categories in both cases were vegetables, fruits, drinks, frozen food and ready meals. Nouns shorter than four or longer than eleven characters were excluded, while no more than four nouns in any category had the same initial letter. The category name was shown in the title bar at the top of the menu.

Experimental design: Each of the two experiments followed a within-subjects design and was planned to fit into a 1 h session. Subjects were informed that

Table 1: Selection frequency of small and large menu items and their distribution

	Distribution			Distribution			Distribution
Item	Small	Large	Item	Small	Large	Item	Large
1	0	0	11	2	4	21	2
2	0	0	12	4	6	22	4
3	4	0	13	10	0	23	0
4	8	0	14	12	0	24	8
5	0	6	15	2	0	25	0
6	4	8	16	20	8	26	10
7	0	4	17	8	0	27	12
8	10	6	18	-	4	28	6
9	4	0	19	-	10	29	2
10	12	0	20	-	0	-	-

the menu conditions were divided into two blocks, where block 1 consisted of a 50 item sequence selection and block 2 consisted of the identical 50 item sequence to block 1. Between the two blocks, subjects were given a 2 min break. For the adaptable condition, subjects were allowed to take extra time during the break to customize their menus if they wished to do so. This was their only opportunity to customize.

Selection frequency: Table 1 shows the distribution of the selection frequencies used in the two experiments. The numbers in the first, fourth and seventh rows of the table indicate the vertical position of an item as number of places from the top. The second and fifth rows show how many times an item would occur in 100 selections for the small menu, while the third, sixth and eighth rows show how many times an item would occur in 100 selections for the large menu. The distributions for small and large menus were adapted from the literature with some modification[13].

Procedure: First, subjects were randomly assigned to different orders of conditions depending on the order of arrival, then a questionnaire was used to obtain information on user demographics, education and computer experience. Before starting each menu condition, subjects were given a recorded tutorial. In the experiment, the subjects performed the five conditions in a predetermined order given by the experimenter. A condition comprised of two task blocks, each of which contained 50 selections. Therefore, each subject performed a total of 500 selections. First, subjects were asked to choose the menu condition according to the order given by the experimenter. The first task block began when the subjects clicked the 'Start' button. Next, a target item was displayed on the screen and subjects were asked to select the same item from the pull-down menu as quickly and accurately as possible. If the wrong item was clicked a cross symbol appeared on the screen. The second target item appeared once the target item had been selected. When a subject selected the correct item, the menu was disabled for 1 s before the next item. Time between the presentation of the target item and the correct selection was recorded, as well as the number of errors (incorrect selections). In the adaptable and adaptive/adaptable minimized menus, subjects were told that they could change the positions of the items if they wanted to do so after the first block. In addition, the time required by each subject to customize the adaptable menu was recorded. In block 2, item positions remained as they were at the end of block 1. The primary reason for this was to measure the effects of the changes made in block 1, while subjects performed differently. In other words, if subjects had begun block 2 from the same point that they had begun block 1, the result would not have been expected to change. On the other hand, menu design remained as it was, to unify menu conditions across all blocks. For example, in highlighted and mixed-initiative menus the highlighted items would fade away. Finally, a feedback questionnaire was used to rank the menu conditions, to assess subjects' satisfaction and to record any additional comments.

RESULTS

The aim of this study is to combine the two results[24,25] to examine the affects of menu size on user performance.

Selection time: A 2×5 (screen size x menu conditions) repeated measures ANOVA was performed. Mauchly's test indicated that the assumption of sphericity had been violated for the main effects of the menus ($X^2(2) = 16.98$, $p<0.05$) and for the interaction between menu type and size ($X^2(2) = 25.47$, $p<0.05$). Therefore, degrees of freedom were corrected using Greenhouse-Geisser estimates of sphericity (E = 0.86 for the main effect of menus and 0.81 for the main effect of the interaction between menu type and size). All effects were reported as significant at $p<0.05$. There was a significant main effect of menu type on ratings of the menus: $F(3.43, 202.52) = 3.34$. Comparisons revealed that adaptable menus were faster than adaptive split menus: $F(1,59) = 5.41$, $r = 0.29$. No other significant differences were found when comparing minimized menus to the baseline (adaptable) ($F(1,59) = 0.73$, $r = 0.11$), highlighted menus to adaptable ($F(1,59) = 0.16$, $r = 0.05$) or mixed-initiative menus to adaptable ($F(1,59) = 2.01$, $r = 0.18$). In addition, there was a significant main effect of size on menu ratings ($F(1.0, 59.0) = 13.107$) and a significant interaction effect between menu type and size ($F(1, 59) = 13.11$).

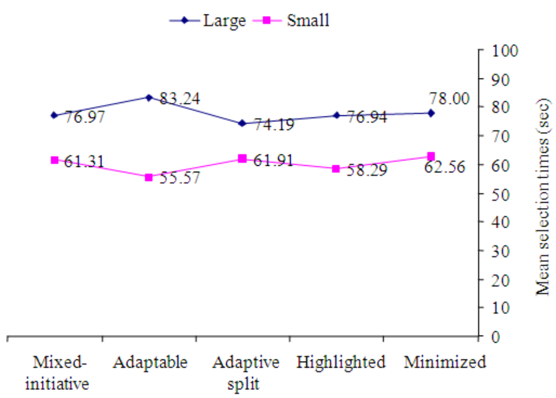

Fig. 3: Mean selection times for small and large menus, block 1

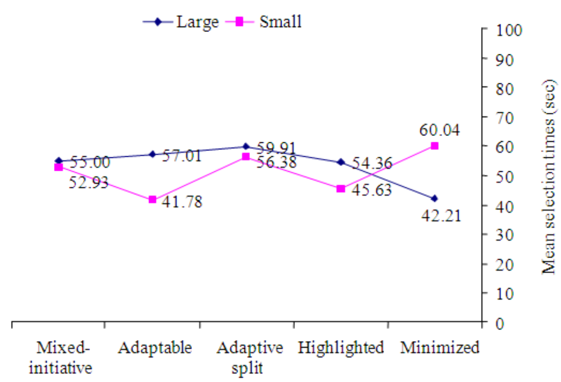

Fig. 4: Mean selection times for small and large menus, block 2

This indicates that size had different effects on participants' performance depending on which type of menu was used. To break down this interaction, all menu types were compared to the baseline (adaptable menu) and both sizes to the baseline (small). This revealed significant interactions when comparing small with large menus for minimized ($F(1,59) = 40.22$, $r = 0.64$), highlighted ($F(1,59) = 7.94$, $r = 0.34$), split ($F(1,59) = 22.81$, $r = 0.52$) and mixed-initiative menus ($F(1,59) = 28.47$, $r = 0.57$), against the baseline menu (adaptable).

Figure 3 and 4 shows that when using small menus in block 1, subjects were significantly faster under the adaptable approach than all other menus, whereas this approach was the slowest for large menus. The adaptive split approach was significantly slower than both adaptable and adaptive highlighted approaches in small menus, but considerably faster than all others for large menus. In block 2 of the small menu experiment, subjects were again significantly faster when using the

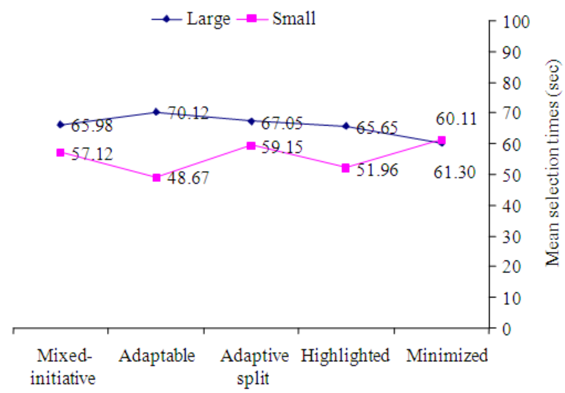

Fig. 5: Mean overall selection times

adaptable approach than all other approaches, whereas the best approach for the large menus in block 2 was the adaptive minimized approach, which was considerably faster than other approaches (42.21 sec). In addition, the adaptive split menu was the slowest approach in large menus, with (59.91 sec), whereas for small menus this was the second least efficient condition (56.36 sec). In block 1 of the large menu experiment, the adaptable approach was the least efficient (83.24 sec). By contrast, this approach was the most efficient in small menus (only 55.57 sec). For large menus, the split approach was the most efficient (74.19 sec), whereas for small menus this was the second least efficient condition (61.91 sec), the minimized approach being the least efficient. On the other hand, in block 2, the adaptable condition was the most efficient approach to small menus, but the second least efficient condition in large menus (57.01 sec), while the split menu was the least efficient (59.91 sec). There was no significant difference in efficiency between the adaptive split and mixed-initiative conditions in either small or large menus. Furthermore, in block 2 of the large menu experiment, the minimized approach was surprisingly the most efficient condition, even more than its counterpart in the small menu experiment.

Figure 5 depicts the overall results for both menus and shows that some approaches behaved in the opposite way to others. By way of illustration, the adaptable approach was the most efficient in small menus, but the slowest one in large menus, while the opposite was true of the minimized approach, which was the most efficient in large menus and the slowest in small menus. Interestingly, there was no significant difference in this approach for the mean overall selection times between large and small menus.

Table 2: Frequency of user errors

Menu Block	Small 1	Small 2	Large 1	Large 2	Sum
Mixed-initiative	14	15	15	16	60
Split	11	17	13	11	52
Highlight	19	12	11	10	52
Adaptable	16	8	13	21	58
Minimized	16	16	15	7	54
Total	76	68	67	65	276

Table 3: Approaches utilized in each block

Menu	Approach Block 1	Approach Block 2
Highlighted	Adaptive	Adaptable
Adaptable	Traditional	Adaptable
Minimized	Adaptive	Adaptable
Mixed-initiative	Mixed-initiative	
Split	Adaptive	

Error rate: Table 2 shows the total number of errors for all subjects for each menu condition. An error was recorded when a subject clicked an item that was different from the target. Each cell contains the number of errors the subjects made in 3000 selections (50 selections $\times$ 2 blocks $\times$ 30 subjects) in each condition. It can be seen that customization helped to reduced the number of errors in the adaptable small menu and minimized large menu, whilst it increased between blocks 1 and 2 in the adaptable large menu. This increase is can be explained by the fact that the additional content of the large menu caused confusion. In small menus, 50% of the errors were eliminated in block 2 in the adaptable approach, while improvement was slightly less marked for the highlighted menu. The mixed-initiative approach differed from all others in that the number of errors remained largely constant for both blocks and both menu sizes.

DISCUSSION

The aim of this study is to compare the three personalization approaches: Adaptive, adaptable and mixed-initiative. For small menus, the results show that the adaptable menu was the fastest in both blocks (Fig. 3 and 4), from which we may conclude that adaptable menus are more efficient than adaptive ones. A possible explanation for this is that the size of the menu helps subjects to remember the position of items. This can be confirmed by the observation that some subjects preferred not to customize the adaptable menus. This finding is in agreement with those of Findlater and McGrenere[6] and of Park *et al*.[23], who report that adaptable menus were more efficient than adaptive ones. Although these results differ from some published studies[13], they are consistent with the finding that an adaptive split menu was faster than a highlighted one. In addition, is different from the large menu results[24]. Moreover, in mixed-initiative menus, there were two reasons for subjects’ uncertainty. First, the mixed-initiative menu repeatedly updates the items in the recently-used list. The second reason is that subjects need to choose to display either the recently or frequently-used items. By contrast, in the minimized menu the recency technique is neglected and only the frequency is taken into account. These drawbacks seem to limit the effectiveness of both menu types. In large menus, the results show that the adaptive split menu was the fastest in block 1, although it was surprisingly the slowest in block 2. This leads us to conclude that using adaptation as a regular technique might strain subjects. This finding is in agreement with that of Sears and Shneiderman[13], who report that the adaptive split menu was faster than the adaptable one. Although these results also differ from some published studies[6], they are consistent with the finding that adaptable menus were more efficient than adaptive ones. In addition, is different from the small menu results[25]. Importantly, some menus utilized different approaches from one block to the other. For example, the adaptable menu employed the traditional approach in block 1, as subjects did not adapt it until the second block. Similarly, the highlighted and minimized menus utilized adaptive techniques in block 1 and adaptable ones in block 2. The difference here is that in highlighted menus, item positions remain the same, while in minimized ones they change. As for the mixed-initiative and adaptive split menus, these maintained the same approach in both blocks. Present study is different from others because our comparison involved a combination of different approaches (Table 3).

The traditional approach in block 1 was based on users memorizing item positions; as pointed out by[1], it takes time to memorize the position of all the items and even when the position of frequently used items is known, the menu does not provide any support. However, the results for this approach varied according to menu size: It was the fastest condition for small menus but the slowest for large ones. This confirms that the traditional approach is efficient for small menus but less so as content increases. In block 2, subjects were able to customize the menu by reordering the items or putting frequently used ones at the top of the list. However, they still had to memorize the positions of items in order to customize the menu. Again, this approach was the fastest for small menus, while for large ones it was found to be the second slowest.

Highlighted menus: The highlighted approach required less memorizing of item positions, since the menus provided support by highlighting the position of frequently used items. The results show that because the frequently used items were already known in small highlighted menus, subjects took slightly less time to customize the menu: An average of 8.59 min, compared to 8.89 min for adaptable menus. The difference was much greater with large menus: An average of 5.52 min compared to 11.67 min for the adaptable condition. However, the highlighted small menu had no significant advantage over the adaptable menu in either block 1 or 2 in terms of the selection time. This means that no such advantage is obtained by highlighting frequently selected items, whereas highlighted large menus had an advantage over adaptable menus in both blocks.

Adaptive split menus: The large adaptive split menu was faster than other conditions in block 1, but was surprisingly the slowest in block 2. A possible explanation for this is that the size of the searching area affects subjects' behavior, since in block 1 subjects had to consider the whole menu, whereas in block 2 the frequently clicked items moved to the top of the list and subjects neglected the bottom of the menu. This can be confirmed by observation and interviewing subjects after the experiment. In addition, the results for block 1 show that the small adaptive split menu was very slow. This result is consistent with those of Findlater and McGrenere[10], who report that accessing menu items on a small screen was slower than on a large screen. This may also be the case for searching for items in a small area compared to a large one.

Adaptive minimized menus: In the minimized menu the recency technique was neglected and only the frequency was taken into account. Unfortunately, this design caused subjects to obtain the benefit of the frequently used items only, which seems to have limited the effectiveness of this menu. Therefore, further work needs to be done to establish whether utilizing the recency and frequency techniques would be more beneficial.

Mixed-initiative menus: In mixed-initiative menus there were two reasons for uncertainty among subjects: First, this type of menu repeatedly updates the items in the recently-used list; secondly, subjects must choose to display either recently or frequently-used items. These drawbacks seem to limit the effectiveness of this menu type. There is therefore a definite need to show both recently and frequently used items, while avoiding repeated updates of the items.

Adaptation by users: Subject could easily move items up and down by clicking on the required item and then on an up or down arrow placed above the menu. They were told how to customize and provided with help when needed, since we were interested in the results of customization, not the way in which it was done. However, subjects utilized different criteria for ordering the menu items. The most common approaches were frequency-based and alphabetical ordering. This did not prevent some subjects from using their own criteria. For example, one subject moved the items near the top to the top of the list and items near the bottom to the bottom of the list. However, one of the main objectives of this study was to investigate the effect of different levels of adaptability. Therefore, we conducted in block 2 a comparison of three adaptable menus presented with different types of adaptability: (1) help not provided (that is, adaptable menu), (2) assistance provided by highlighting the frequently clicked items (that is, highlighted menu) and (3) recommendation provided by moving frequently clicked items to the top of the list, followed by a horizontal line separating the recently clicked items and hiding the others (that is, minimized menu). In the small menu experiment, the results for the adaptable menu show that it was more efficient than both the highlighted and minimized menus in block 2. In addition, the highlighted and minimized menus were approximately the same in block 2. It is difficult to explain this result, but it may be related to the fact that when asked in the interview, subjects said that they felt in more control when using the adaptable menu than either the highlighted or minimized ones. For large menus, the minimized type was found to be more efficient than both highlighted and adaptable menus.

The results show that subjects behaved differently towards highlighted and adaptable menus according to their size; for example, they customized large menus less than small ones. In addition, subjects who customized adaptable menus spent more time on large than small ones, while those who customized highlighted menus spent less time on large than small ones. These results may be explained by the fact that highlighting some of the items of a small menu makes it look visually more complex than highlighting the same number in a larger one. It was also found that under the mixed-initiative condition, subjects utilized the frequency and recency techniques more in large than small menus, with respective totals of 120 and 94 selections made by subjects.

Adaptation by the system: A second objective was to investigate the effect of different levels of adaptation. Therefore, we conducted a comparison of three adaptive menus in block 1 presented with different types of adaptation: (1) changes occurring without moving items (that is, highlighted menu), (2) changes made by moving recently and frequently clicked items to the top of the list and leaving the others unchanged (that is, split menu) and (3) changes made by moving only frequently clicked items to the top of the list and leaving the others unchanged (that is, minimized menu). On one hand, the results for the adaptive menus in small conditions show that highlighting the frequently clicked items was more efficient than changing both the recently and frequently clicked items or solely the frequently clicked items. This result may be explained by the fact that subjects preferred fewer changes to occur. On the other hand, the results for adaptive menus showed that changing both the recently and frequently clicked items was more efficient than just changing or highlighting the frequently clicked items.

CONCLUSION

The study reported in this study provides empirical evidence that adaptive, adaptable and mixed-initiative menus have varied impacts depending on menu size. In addition, it shows that menu size affects users' behavior: Subjects were more likely to customize small menus than large ones. In conclusion, this study has thrown up many questions in need of further investigation, such as whether we could mitigate the drawbacks of each condition and then to find out which condition is most usable. It would also be useful to investigate the factors making some of them more successful in one context than another. Therefore, more research needs to be undertaken on this topic to understand these approaches from different perspectives.

REFERENCES

1. Hsi, I. and C. Potts, 2000. Studying the evolution and enhancement of software features. Proceeding of the 16th IEEE International Conference on Software Maintenance, Oct. 11-14, IEEE Xplore Press, San Jose, CA., USA., pp: 143-151. DOI: 10.1109/ICSM.2000.883033
2. Davis, F.D., R.P. Bagozzi and P.R. Warshaw, 1989. User acceptance of computer technology: A comparison of two theoretical models. Manage. Sci., 35: 982-1003. http://www.jstor.org/pss/2632151
3. McGrenere, J. and G. Moore, 2000. Are we all in the same" bloat"? Proceedings of Graphics Interface, May 2000, Montreal, pp: 1-10. http://www.dgp.toronto.edu/~joanna/papers/gi_2000_bloat.pdf
4. McGrenere, J., 2002. The design and evaluation of multiple interfaces: A solution for complex software. The University of Toronto, Toronto. http://people.cs.ubc.ca/~joanna/papers/JMcGrenere_Thesis_DoubleSided.pdf
5. Lee, D.S. and W.C. Yoon, 2004. Quantitative results assessing design issues of selection-supportive menus. Int. J. Ind. Ergon., 33: 41-52. DOI: 10.1016/j.ergon.2003.07.004
6. Findlater, L. and J. McGrenere, 2004. A comparison of static, adaptive and adaptable menus. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Apr. 24-29, ACM Press, Vienna, Austria pp: 89-96. DOI: http://doi.acm.org/10.1145/985692.985704
7. Holzinger, A., M. Kickmeier-Rust and D. Albert, 2008. Dynamic media in computer science education; content complexity and learning performance: Is less more? Educ. Technol. Soc., 11: 279-290. http://eric.ed.gov/ERICWebPortal/custom/portlets/recordDetails/detailmini.jsp?_nfpb=true&_&ERICExtSearch_SearchValue_0=EJ814086&ERICExtSearch_SearchType_0=no&accno=EJ814086
8. Findlater, L., K. Moffatt, J. McGrenere and J. Dawson, 2009. Ephemeral adaptation: The use of gradual onset to improve menu selection performance. Proceeding of the 27th International Conference on Human Factors in Computing Systems, Apr. 4-9, ACM Press, Boston, MA., USA., pp: 1655-1664. http://portal.acm.org/citation.cfm?doid=1518701.1518956
9. Shneiderman, B. and P. Maes, 1997. Direct manipulation Vs interface agents. Interactions, 4: 42-61. http://doi.acm.org/10.1145/267505.267514
10. Findlater, L. and J. McGrenere, 2008. Impact of screen size on performance, awareness and user satisfaction with adaptive graphical user interfaces. Conference on Human Factors in Computing Systems, Proceeding of the 26th Annual SIGCHI Conference on Human Factors in Computing Systems, Apr. 05-10, ACM Press, Florence, Italy, pp: 1247-1256. http://doi.acm.org/10.1145/1357054.1357249
11. Gajos, K.Z., M. Czerwinski, D.S. Tan and D.S. Weld, 2006. Exploring the design space for adaptive graphical user interfaces. Proceedings of the Working Conference on Advanced Visual Interfaces, May 23-26, ACM Press, Venezia, Italy, pp: 201-208. http://doi.acm.org/10.1145/1133265.1133306

12. Greenberg, S. and I.H. Witten, 1985. Adaptive personalized interfaces-A question of viability. Behav. Inform. Technol., 4: 31-45. http://grouplab.cpsc.ucalgary.ca/grouplab/uploads/Publications/Publications/1985-AdaptiveInterfaces.BIT.pdf
13. Sears, A. and B. Shneiderman, 1998. Split menus: Effectively using selection frequency to organize menus. Technical Report CAR-TR-649. http://www.lib.umd.edu/drum/handle/1903/386?mode=simple
14. Gong, Q. and G. Salvendy, 1995. An approach to the design of a skill adaptive interface. Int. J. Human Comput. Interact., 7: 365-384. http://psycnet.apa.org/index.cfm?fa=search.displayRecord&uid=1996-02934-005
15. Robert, T. and P.B. Dermot, 1987. A self-regulating adaptive system. SIGCHI Bull., 18: 103-107. http://doi.acm.org/10.1145/1165387.30867
16. Mitchell, J. and B. Shneiderman, 1989. Dynamic versus static menus: An exploratory comparison. ACM SIGCHI Bull., 20: 33-37. http://doi.acm.org/10.1145/67243.67247
17. Bunt, A., 2007. Mixed-Initiative Support for Customizing Graphical User Interfaces. University of British Columbia, Canada, ISBN: 978-0-494-31745-7, pp: 236.
18. Oppermann, R. and M. Specht, 1999. A nomadic information system for adaptive exhibition guidance. Arch. Museum Inform., 13: 127-138. DOI: 10.1023/A:1016619506241
19. Read, J.C., S. MacFarlane and C. Casey, 2002. Oops! silly me! errors in a handwriting recognition-based text entry interface for children. Proceedings of the 2nd Nordic Conference on Human-Computer Interaction, Oct. 19-23, ACM Press, Aarhus, Denmark, pp: 35-40. http://doi.acm.org/10.1145/572020.572026
20. Al-Omar, K. and D. Rigas, 2008. An empirical study to investigate the efficiency of static, adaptable, adaptive and mixed-initiative environments in e-commerce. Proceedings of the IADIS International Conference on Interfaces and Human-Computer Interaction, July 25-27, InderScience Publishers, IADIS, Amsterdam, Netherlands, pp: 234-238.
21. Al-Omar, K. and D. Rigas, 2008. A platform for investigating effectiveness for static, adaptable, adaptive and mix-initiative environments in e-commerce. Proceedings of the International Conference on E-Business, July 26-29, ICETE ICE-B, Porto, Portugal, pp: 191-196.
22. Rigas, D. and K. Al-Omar, 2008. A platform for investigating user satisfaction for static, adaptable, adaptive and mix-initiative environments in e-commerce. Proceedings of the Saudi International Innovation Conference, (SIIC’08), Leeds, UK., pp: 63-67.
23. Park, J., S.H. Han, Y.S. Park and Y. Cho, 2007. Adaptable versus adaptive menus on the desktop: Performance and user satisfaction. Int. J. Ind. Ergon., 37: 675-684. DOI: 10.1016/J.ERGON.2007.04.006
24. Al-Omar, K. and D. Rigas, 2009. Comparison of adaptive, adaptable and mixed-initiative menus. Proceedings of the International Conference on CyberWorlds, Sept. 7-11, Bradford, UK., pp: 292-297. http://www.computer.org/portal/web/csdl/doi/10.1109/CW.2009.29
25. Al-Omar, K. and D. Rigas, 2009. A user performance evaluation of personalized menus. Proceedings of the International Conference on ICADIWT, London, UK.