On User Behaviour Adaptation Under Interface Change

Citation for published version:
Rosman, B, Ramamoorthy, S, Mahmud, MMH & Kohli, P 2014, On User Behaviour Adaptation Under Interface Change. in *Proceedings of the 19th International Conference on Intelligent User Interfaces.* ACM, New York, NY, USA, pp. 273-278. https://doi.org/10.1145/2557500.2557535

Digital Object Identifier (DOI):
10.1145/2557500.2557535

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Proceedings of the 19th International Conference on Intelligent User Interfaces

On User Behaviour Adaptation Under Interface Change

Benjamin Rosman,* Subramanian Ramamoorthy, M. M. Hassan Mahmud
School of Informatics
University of Edinburgh
Edinburgh, UK

Pushmeet Kohli
Machine Learning and Perception
Microsoft Research
Cambridge, UK

ABSTRACT
Different interfaces allow a user to achieve the same end goal through different action sequences, e.g., command lines vs. drop down menus. Interface efficiency can be described in terms of a cost incurred, e.g., time taken, by the user in typical tasks. Realistic users arrive at evaluations of efficiency, hence making choices about which interface to use, over time, based on trial and error experience. Their choices are also determined by prior experience, which determines how much learning time is required. These factors have substantial effect on the adoption of new interfaces. In this paper, we aim at understanding how users adapt under interface change, how much time it takes them to learn to interact optimally with an interface, and how this learning could be expedited through intermediate interfaces. We present results from a series of experiments that make four main points: (a) different interfaces for accomplishing the same task can elicit significant variability in performance, (b) switching interfaces can result in adverse sharp shifts in performance, (c) subject to some variability, there are individual thresholds on tolerance to this kind of performance degradation with an interface, causing users to potentially abandon what may be a pretty good interface, and (d) our main result – shaping user learning through the presentation of intermediate interfaces can mitigate the adverse shifts in performance while still enabling the eventual improved performance with the complex interface upon the user becoming suitably accustomed. In our experiments, human users use keyboard based interfaces to navigate a simulated ball through a maze. Our results are a first step towards interface adaptation algorithms that architect choice to accommodate personality traits of realistic users.

Author Keywords
Input and interaction technologies; usability research; usability testing and evaluation; user interface design.

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

*Benjamin Rosman is also affiliated with the Mobile Intelligent Autonomous Systems group at the CSIR, South Africa.

IUI'14, February 24–27, 2014, Haifa, Israel.

http://dx.doi.org/10.1145/2557500.2557535

INTRODUCTION
Interfaces are an important component of computing applications that require interaction between man and machine, ranging from text entry based search, to configuration dialogs to emerging natural user interfaces that provide fully immersive experiences in video games, etc. In typical sequential decision making tasks, users execute sequences of atomic actions that may be configured in various ways. A set of such atomic actions defines an interface.

Given an action set/interface, and many tasks that must be performed with it, users eventually learn policies that map task specifications to sequences of actions. After significant experience, the user typically reaches a level of performance that characterises the efficiency of that interface. At this point, if a new interface were to be introduced, how should the user respond? A fully rational user might be expected to evaluate the expected efficiency of this new interface and adopt it if it could yield a better long term efficiency. In many practical settings, users seem inefficient in making this choice. A key factor here is that of learnability – evaluating the efficiency of an interface takes time, and a boundedly rational user (e.g., one with limited patience) could well arrive at a different evaluation in typical usage [13] [1]. The main aim of this paper is to study this phenomenon through empirical experiments.

The problem of devising a good alphabet of actions has been successfully addressed as one of combinatorial optimisation, e.g., in [5]. Other related work addresses design space exploration [4], determining optimal parameters within static user models [8], adaptively determining the best interface based on context variables such as mobile/desktop [6], [1] and other forms of personalization [10]. In contrast to these works that focus on estimating context from which it is clear what interface to present, our focus is on the temporal nature of the process – on how people learn to use interfaces – based on which we wish to determine how the user herself might choose an interface, which again is based on her own prior experience and private evaluation.

Our experiments are designed to test hypotheses regarding learnability of interfaces. We posit that users with limited patience and diverse prior experience will not only have different levels of initial success with different interfaces but also,

[1] See e.g., http://blogs.msdn.com/b/b8/archive/2012/05/18/creating-the-windows-8-user-experience.aspx, http://www.pcworld.com/article/2012024/the-windows-8-ui-how-do-interface-and-usability-experts-rate-all-the-changes.html, http://www.addictivetips.com/windows-tips/is-adapting-to-windows-8-and-its-metro-ui-as-hard-as-it-seems/

given the temporal nature of their habituation [7] with interfaces and limited patience, they may incorrectly evaluate efficiency and prematurely abandon good interfaces. Finally, we hypothesize that this adoption behaviour may be improved by presenting intermediate interfaces that mitigate the initial shock and encourage learning towards the better performance level.

Our hypotheses are directly relevant to practical concerns, such as the need to minimise change aversion [12], [9], consumer choices regarding new products [11], etc. Also, we add to a literature on modelling user adaptation [14], [2], [15], [3] by empirically characterising dynamic choice behaviour.

Our Contributions

Our experiments show that users adapt to interfaces, such that their performance with a particular interface improves with time. We then show that an interface change can dramatically degrade user performance even though the new interface might be better theoretically. This performance slowly improves as the user adapts to the new interface while interacting with it repeatedly. Finally, we show that a well-designed intermediate interface can dramatically reduce the degradation in performance and can also lead to the user learning more quickly to interact with the final interface.

DOMAIN AND PROTOCOL

The aim of our experiments is twofold. Our first goal is to study how user performance and preference over interfaces varies with complexity of the interfaces. Our second goal is to test if the user's experience can be shaped to use complex but more powerful interfaces while minimising the performance hit during training by using an interface of intermediate complexity.

To that end, we implemented a system which required a user to repeatedly perform a task, using different interfaces. Each task executed by the users of our system involves navigating a ball through a simple maze of 1000×1000 pixels to a goal location in the shortest possible time, as shown in Figure 1. Obstacles block the motion of the ball, and each task ends when the goal is reached. Each user performs a number of such tasks in a set of mazes, where the mazes are standardised across users, and the time taken for each task is recorded. We used 20 such mazes, where the mazes were generated randomly initially and then fixed for all experiments.

For the above tasks, we designed three different interfaces for controlling the ball. The control scheme for the mazes is described in Table 1. The interfaces were designed to represent different tradeoffs between power and ease of use. The first interface, *I1*, is the simplest to use and the least powerful. It consists of the arrow keys, where pressing the arrow key moves the ball in the expected direction by 20 pixels. The second interface, *I2*, is in the middle in terms of power and difficulty. It requires using two counter-intuitive keys and then the arrow key to move in the expected direction by 80 pixels. The third interface, *I3* is the most difficult to use and also the most powerful. To move in a direction, the user needs to press four counter-intuitive keys in sequence and then her ball is moved by 120 pixels. We use the convention that if

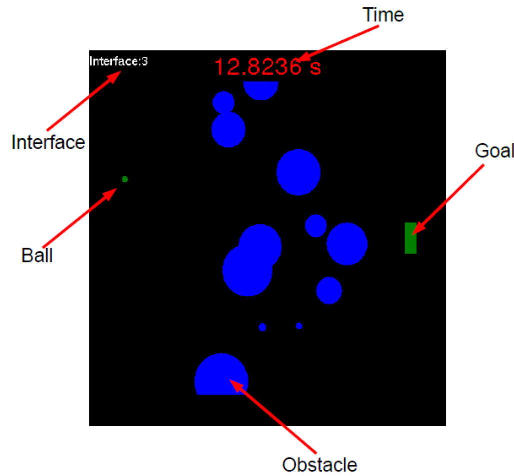

Figure 1. Depiction of the domain used in our experiments. The user navigates a ball to a goal as quickly as possible, while avoiding obstacles.

Key	Direction	Distance
Interface 1		
Key	**Direction**	**Distance**
Left-arrow	Left	Short (20 pixels)
Right-arrow	Right	Short (20 pixels)
Up-arrow	Up	Short (20 pixels)
Down-arrow	Down	Short (20 pixels)
Interface 2		
Key	**Direction**	**Distance**
q h Left-arrow	Left	Medium (80 pixels)
w b Right-arrow	Right	Medium (80 pixels)
u g Up-arrow	Up	Medium (80 pixels)
y f Down-arrow	Down	Medium (80 pixels)
Interface 3		
Key	**Direction**	**Distance**
q h h w	Left	Long (120 pixels)
w b k f	Right	Long (120 pixels)
u g o c	Up	Long (120 pixels)
y f r e	Down	Long (120 pixels)

Table 1. The three interfaces. Interfaces 2 and 3 require sequences of key presses to execute the desired motion. Note that the first two keys for each direction match in interfaces 2 and 3.

the desired motion results in hitting an obstacle, that motion is prevented.

Given the above, the effective distance moved per key press is 20, 26.67 and 30 pixels for *I1*, *I2* and *I3* respectively. Consequently, *I3* is the interface which can achieve the lowest possible times in these navigation tasks. On the other hand, this is also the interface which requires the most learning to be able to issue the commands quickly, as using this interface requires the additional cognitive load of remembering the key presses, or looking them up as needed. Alternatively, *I1* is simple and intuitive to any computer user. Finally, *I2* lies in between *I1* and *I3* in terms of simplicity and intuitiveness.

Now, our goal is to study if we can shape a user to smoothly use a powerful and complex interface, given that she is used to a simpler interface. So, we designate *I1* to be the *start interface*, *I3* to be the *target interface* and *I2* to be the *interme-*

diate interface. The conjectures outlined in the introduction now may be translated as follows. With practice, users will be most effective when using interface *I3*, but at the same time will initially pay a price in terms of decreased performance and increased difficulty when moving from the start interface *I1*. At the same time by using the interfaces *I1*, *I2* and *I3* in sequence, the performance loss and difficulty may be reduced. In the next section we describe experiments to test these conjectures.

EXPERIMENTS AND RESULTS

We performed five sets of experiments with a number of users. Each user was allocated to only one of the sets to ensure that the results were not corrupted by a long-term memory effect. In each experiment the user was required to use different interfaces to solve the navigation tasks according to a given schedule. The schedules are as follows:

1. **Baseline 1:** *I3* for 20 tasks, and then *I1* for 20 tasks.
2. **Baseline 2:** *I1* for 20 tasks, and then *I3* for 20 tasks.
3. **Performance Variability:** *I1* for 20 tasks, and then *I3* for 20 tasks, and again *I3* for 20 tasks.
4. **Intermediate Interface:** *I1* for 20 tasks, and then *I2* for 20 tasks, and then *I3* for 20 tasks.
5. **User Perception:** Repeat 20 times: a training task with *I1*, a training task with *I3*, and then the user's choice of *I1* or *I3* such that time taken is minimised.

We discuss each of these experiments in detail in the next four sections. Briefly, the purpose of each of these experiments is as follows. The two Baseline experiments establish the baseline user performance with the start and target interfaces when used in either order. The Performance Variability experiments establish what happens when we switch from the start to the target interface after some time and keep using it. The Intermediate Interface experiments show user performance when the intermediate interface *I2* is used to bridge the start and target interface. Finally, the User Perception experiment gives insight into which interfaces users prefer over time given prolonged experience with the interfaces. In the following, we assume the *frustration level* of a user is proportional to the time taken for a trial, and further that after some such time she would give up on an interface. We do not directly observe this in our experiments, but conjecture that it exists and state some of our conclusions based on this.

BASELINE PERFORMANCE

First we establish how using different interfaces for the same underlying action space affects performance. Here we provide a baseline performance for users with the simple interface *I1* and the powerful but complicated interface *I3*.

Eight users were tasked with 20 trials using each interface, four of which started with *I3* (Baseline 1) and the other four with *I1* (Baseline 2), so as to remove any effect that the one may have on performance with the second. The results of Baseline 1 are shown in Figure 2, and Baseline 2 in Figure 3.

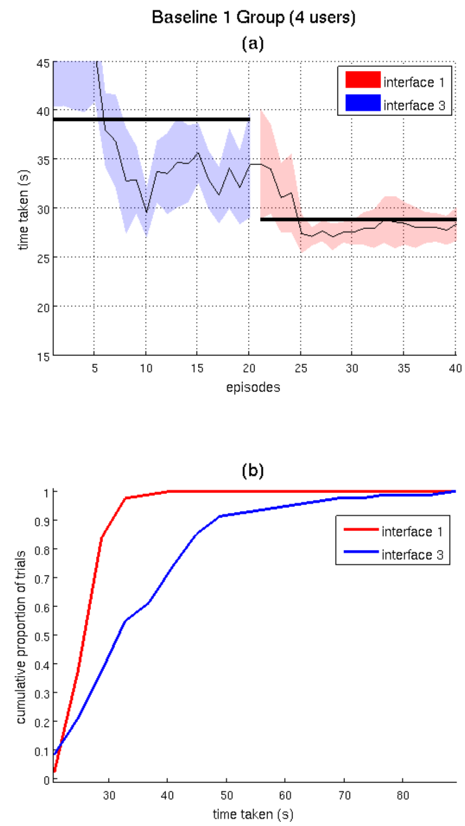

Figure 2. Results for Baseline Group 1. Figure (a) shows the averaged results when interface *I3* is used and then *I1*, for the same sequence of 20 **mazes. The thin black line is the average time taken to goal in the last** 5 **mazes. The thick line gives the average over all the** 20 **mazes. This establishes the baseline for using the interfaces *I3* and *I1* in sequence. Figure (b) shows the cumulative distribution of the time taken to complete the tasks. This shows that without training *I1* is better than *I3*, because for any threshold $x > 21$s for the frustration level of the user, *I3* would breach the frustration for a greater number of tasks than *I1*.**

The results from these groups show that before the user has become familiar with the complex interface, both the mean performance and variance of *I3* are significantly higher than *I1*, although faster times are indeed possible with *I3*.

Note that there is some natural variance in the times taken for the various trials with the same interface. This is because the mazes were randomly generated, and so some were particularly easy or difficult for each interface. We did however standardise these, and each user across the entire study was presented with the same sequence of random mazes, for each interface they used.

PERFORMANCE VARIABILITY

Results in the previous section showed the performance of users in the first 20 trials. As was previously established, *I3* is more difficult for the user, requiring learning of the interface. We thus provided users with a second batch of 20 trials, such that the first would constitute training with this interface. These results are shown in Figure 4.

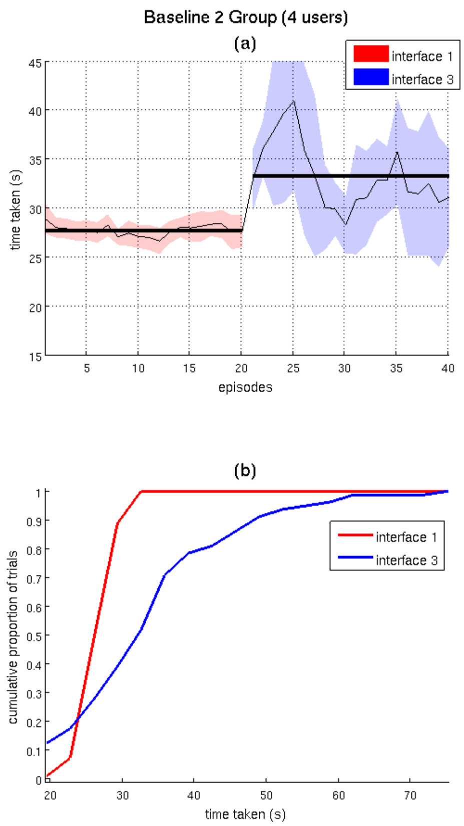

Figure 3. Results for Baseline Group 2. Figure (a) shows the averaged results when interface *I1* is used and then *I3*, for the same sequence of 20 mazes. The thin black line is the average time taken to goal in the last 5 mazes. The thick line gives the average over all the 20 mazes. This establishes the baseline for using the interfaces *I1* and *I3* in sequence. Additionally, note that the performance with *I3* is lower in this task than in Figure 2 (a). We believe that this is because the user has a chance to become familiar with the environment. On the other hand, the performance of *I1* is the same – which shows that familiarity with the domain is not especially beneficial for *I1* and so it is indeed very simple. Figure (b) shows the cumulative distribution of the time taken to complete the tasks. This shows that, again, without training *I1* is better than *I3*, because for any threshold $x > 25$s for the frustration level of the user, *I3* would breach the frustration for a greater number of tasks than *I1*.

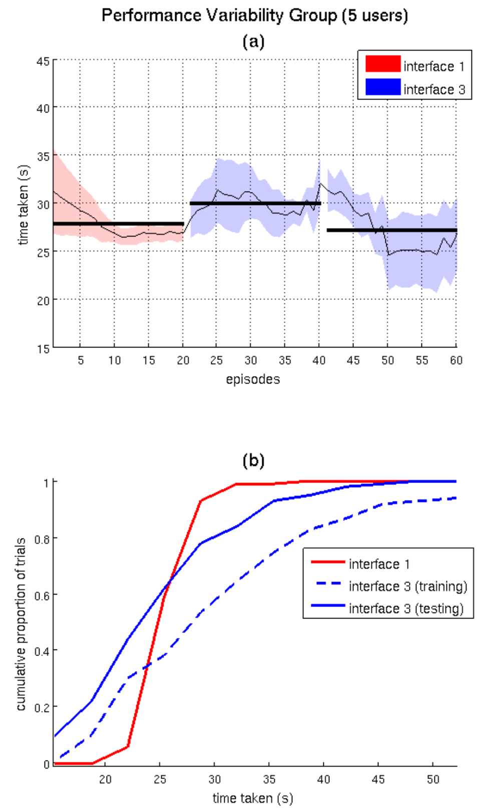

Figure 4. Results for Performance Variability Group. In Figure (a) the thick and thin lines are the same as before. However, now each *I1* is played for the fixed 20 mazes and then *I3* for the fixed 20 mazes, twice in a row – once as a training batch, and then as the testing batch. This figure shows the effect of using *I3* for an extended period of time after using *I1*, and shows that while the user eventually learns to perform much more effectively with *I3* than with *I1*, the spike during the middle 20 mazes shows that there is a performance penalty for this. Figure (b) shows that the frustration level of the user is breached earlier for the training batch of *I3* than for the testing batch of *I3*, which ultimately outperforms *I1*.

As can be seen, after this training phase the users are able to achieve a faster mean time with *I3* than with *I1*. These is however a significant initial spike in performance times during the training phase, before the users consistently outperform their times on the simpler interface. The effect of the training is that both the mean and the variance of the trial times decrease.

INTERMEDIATE INTERFACES

As shown in the previous section, a user with a simple interface can be given a more complicated interface and ultimately achieve better performance. There is however a spike in time taken to complete tasks while the user adapts to the new interface. In Figure 5, we demonstrate that with the correct intermediate interface, a user can be gradually shifted to the more complicated interface, without inducing a spike in time taken.

The adoption of Interface 2 allows the user an easier transition to the full Interface 3, with only a small temporary loss in performance. Interface 2 has a slightly lower mean than Interface 1, with a minimal performance spike. Because Interfaces 2 and 3 are similar, there is also a minimal performance spike when the users make the transition to the final interface.

The key concept here is that although Interface 2 does have the ability to achieve better times than Interface 1, its strength lies mainly in bridging the cognitive divide between the simple and complicated interfaces, and that it accelerates the learning process for the user.

USER PERCEPTION

In real applications, if a user suffers a loss in performance for more than a few episodes, the threshold being subject spe-

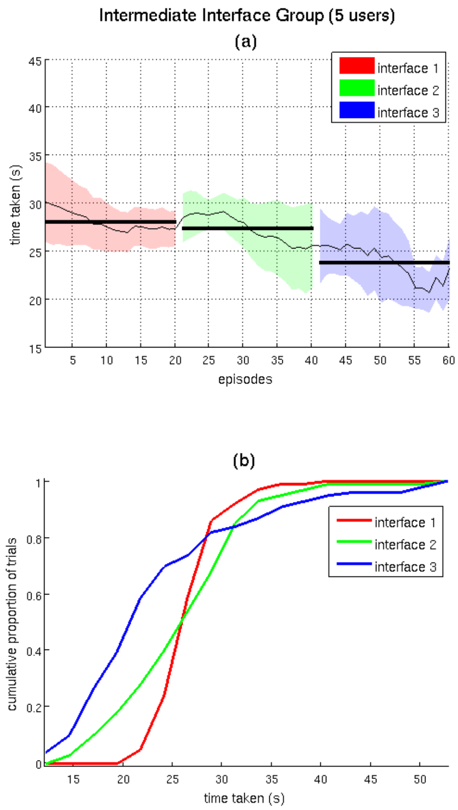

Figure 5. Results for Intermediate Interface Group. In Figure (a) the thick and thin lines are the same as before. Now the interfaces *I1*, *I2* and *I3* are used in sequence for 20 mazes each. *I2* is used as a a bridge to ease transition from the simple interface *I1* to the complex and powerful *I3*. Compared with Figure 4 (a), the user suffers a less severe performance penalty during the middle training phase, while ultimately achieving similar final performance with *I3*. Hence, using this intermediate bridge interface is highly beneficial. Figure (b) also shows a performance improvement over using *I3* as an intermediary.

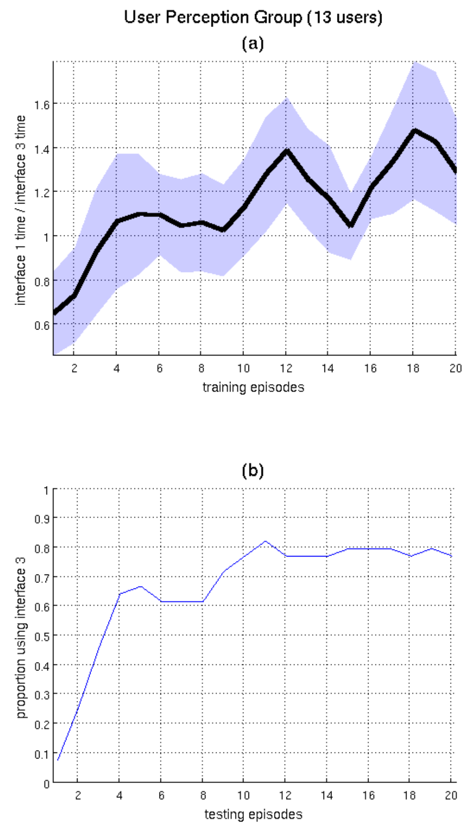

Figure 6. Results for User Perception Group. Figure (a) shows the ratio of times taken by the user during the testing episodes (recall that training and testing is interleaved, and during the training episode the user is trained with both *I1* and *I3*, and then during testing episode, given the choice of using *I1* or *I3*). Eventually, the user does improve with *I3* but this only starts stabilising after episode 10. Figure (b) shows the user preference as a proportion of time they chose interface *I3* during the testing phase. In agreement with Figure (a), this shows that within about 4 testing episodes, the proportion rises above 0.5 and then reaches the maximum 0.8 at about 10 episodes. Hence, this shows that users do eventually realise that using *I3* is the better option, but not quickly.

cific, when switching to a new and potentially better interface (as shown in Figure 4), that user is likely to abandon the new interface (or the application itself) entirely out of frustration.

We observe the effects of this in the User Perception Group. Figure 6 (a) shows the learning curve of human subjects with Interface 3 compared to Interface 1. Note that this ratio only climbs above 1.0 after 9 mazes, corresponding to the point at which users are on average more adept at Interface 3 than Interface 1, in terms of achievable times. The subsequent performance dip around episode 15 is due to mazes which were relatively more difficult for Interface 3.

Figure 6 (b) depicts the proportion of users which chose Interface 3 over Interface 1, in testing mazes. This corresponds to the percentage of users prepared to continue using the more complex interface. During the first 8 mazes, up to 40% of users would rather use the simpler interface, which is indicative of a threshold at which they would abandon the interface. It is only after 10 episodes that about 80% of the users are satisfied with the more powerful interface.

DISCUSSION

Having presented empirical results from our experiments, we now return to our hypotheses outlined in the introduction – what implications can we draw from the data and what does it say about our conjectures?

- Experiments in Figure 4 show how the user's performance degrades when they switch from an elementary interface to a more complex interface, defined by an unfamiliar action set that must be memorised. However, we see that if they stick to this interface setting, they achieve a better performance level after the initial learning period. Although the initial performance degradation is expected from common sense, the fact that users may never realise the true potential of the better interface is counter to standard assump-

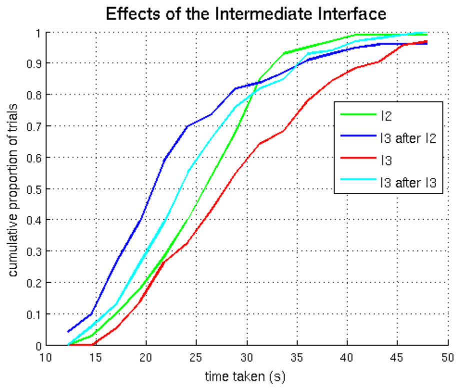

Figure 7. Comparing learning in sequences of interfaces. This figure shows how the user is more effective (more tasks solved within a specified time) in interface *I3* after going through an intermediate interface, versus a control setting where they had longer experience with the same *I3* interface (hence benefitted from learning), without any shaping with the intermediate interface.

tions of expected-value based optimisation procedures that might inform interface optimisation.

- The experiments in Figure 6 show how the *perception* of the efficiency of each interface evolves over time. Although the quantitative threshold varies by user, it is clear that it takes many episodes before the user feels sufficiently comfortable with the initially unfamiliar interface to confidently choose it and realise the potential benefits. If users are impatient, in that they have a short horizon within which they need to see improved performance, they would mistakenly estimate the efficiency of the better interface and prematurely abandon it.

- Experiments in Figure 5 present a way out. If an intermediate interface were available that alleviates the initial loss in performance, the dual goals of lowering performance degradation and eventually achieving the better performance level could be achieved. This is clearly evident from the comparison in Figure 7.

CONCLUSION

The work presented in this paper represents an attempt to study temporal aspects of learning to use an interface. Many approaches to devising optimal interfaces and to adapting interfaces to context are implicitly based on a decision theoretic approach to maximising expected utility. As we show in empirical experiments, the optimal decision is made more complex by the way in which users learn and the extent to which history can play a role in their choice behaviour. This calls for a more refined user model that informs the optimisation process. Our experiments shed light on what attributes need to be captured in such a model. Moreover, we present empirical evidence to the effect that the problems raised by learning and boundedly rational user choices can be mitigated by introducing intermediate interfaces. One way in which this idea can be applied more generally is in interfaces that are continuously parametrisable, for example based on gestures or wearable sensors. Our experiments, such as in Figures 5 and 6, are precursors to a learning algorithm that can make choices regarding sequences of interface settings in an automated fashion. Designing and implementing such algorithms is an area for future work.

ACKNOWLEDGEMENTS

The authors gratefully thank the reviewers for their insightful and constructive comments.

This work has taken place in the Robust Autonomy and Decisions group within the School of Informatics. Research of the RAD Group is supported by the European Commission (TOMSY Grant Agreement 270436, under FP7-ICT-2009.2.1 Call 6).

REFERENCES

1. Anderson, C. R., Domingos, P., and Weld, D. S. Adaptive web navigation for wireless devices. In *In Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence* (2001).
2. Beaudry, A., and Pinsonneault, A. Understanding user responses to information technology: a coping model of user adaption. *MIS Quarterly 29(3)* (2005), 493–524.
3. Dessart, C.-E., Genaro Motti, V., and Vanderdonckt, J. Showing user interface adaptivity by animated transitions. In *Proceedings of the 3rd ACM SIGCHI symposium on Engineering interactive computing systems*, ACM (2011), 95–104.
4. Gajos, K. Z., Czerwinski, M., Tan, D. S., and Weld, D. S. Exploring the design space for adaptive graphical interfaces. In *Proceedings of the Working Conference on Advanced visual interfaces* (2006).
5. Gajos, K. Z., Wobbrock, J. O., and Weld, D. S. Improving the performance of motor-impaired users with automatically-generated, ability-based interfaces. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ACM (New York, NY, USA, 2008), 1257–1266.
6. Hartmann, M., and Schreiber, D. Proactively adapting interfaces to individual users for mobile devices. In *Adaptive Hypermedia and Adaptive Web-Based Systems* (2008).
7. James, W. *Habit*. 1890.
8. Kohli, P., Nickisch, H., Rother, C., and Rhemann, C. User-centric learning and evaluation of interactive segmentation systems. *International Journal of Computer Vision 100(3)* (2012), 261–274.
9. Lavie, T., and Meyer, J. Benefits and costs of adaptive user interfaces. *International Journal of Human-Computer Studies 68*, 8 (2010), 508–524.
10. Mourlas, C., and Germanakos, P. *Intelligent User Interfaces: Adaptation and Personalization Systems and Technologies*. Information Science Reference, 2008.
11. Ram, S., and Sheth, J. N. Consumer resistance to innovations. *Journal of Consumer Marketing 6(2)* (1989).
12. Sedley, A., and Müller, H. Minimizing change aversion for the google drive launch. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems* (2013).
13. Sinofsky, S. Creating the windows 8 user experience. http://blogs.msdn.com/b/b8/archive/2012/05/18/creating-the-windows-8-user-experience.aspx, 2012.
14. Smith, C. L., and Kantor, P. B. User adaptation: good results from poor systems. In *Proceedings of the* 31^{st} *Annual International ACM SIGIR conference on Research and Development In Information Retrieval* (2008).
15. Zajonc, R. B. Attitudinal effects of mere exposure. *Journal of Personality and Social Psychology 9* (1968).