New Review of Hypermedia and Multimedia, Vol. 10, No. 1, June 2004, 5–29

Usable adaptive hypermedia systems

T. TSANDILAS*[†] and M. C. SCHRAEFEL[‡]

[†]Department of Computer Science, University of Toronto, Toronto M5S 3G4, Ontario, Canada; Email: fanis@cs.toronto.edu

[‡]Department of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK; Email: mc@ecs.soton.ac.uk

What did you come in to look at?
If you have any order to give me it's my duty to carry it out, he answered, after another silent pause, with a slow, measured lisp, raising his eyebrows and calmly twisting his head from one side to another, all this with exasperating composure.

Notes from the Underground, Fyodor Dostoyevsky

Adaptive interfaces have received much criticism because adaptation and automatic assistance generally contradict the principles of direct-manipulation interfaces. In addition, their success depends highly on the ability of user models to capture the goals and needs of the users. As the construction of user models is often based on poor evidence, even the most advanced learning algorithms may fail to infer accurately the user goals. Previous research has put little emphasis on investigating usability problems of adaptive systems and developing interaction techniques that could resolve these problems. This paper examines these problems and presents an interaction model for adaptive hypermedia (AH) that merges adaptive support and direct manipulation. This approach is built upon a new content adaptation technique that derives from fisheye views. This adaptation technique supports incremental and continuous adjustments of the adaptive views of hypermedia documents and balances between focus and context. By combining this technique with visual representations and controllers of user models, we form a twofold interaction model that enables users to move quickly between adaptation and direct control. Two preliminary user studies exhibit the strengths of our proposed interaction model and adaptation technique. Future extensions to our work are outlined based on the weaknesses and limitations that the studies revealed.

Keywords: Adaptive hypermedia; Usability; Predictability; User control; Focus+context; Fisheye views

1. Introduction

The usability problems of adaptation and automatic assistance have long been noted (Shneiderman and Maes 1997) mainly by researchers in the community of human–computer interaction (HCI). A main argument against adaptive systems is that they usually violate usability principles that have been established for

*Corresponding author.

New Review of Hypermedia and Multimedia
ISSN 1361-4568 (print)/ISSN 1740-7842 (online) © 2004 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/13614560410001728137

direct-manipulation interfaces. Partly because of these problems, as well as the lack of thorough evaluation studies, intelligent interfaces have not succeeded in proving their usefulness. As Hook (2000) notes, the very few intelligent interfaces that have succeeded commercially have either performed very simple adaptations based on limited knowledge about the user or based adaptation on simple user actions rather than trying to infer complex user models. On the other hand, user interfaces become more and more complex while the problem of information overload burdens the use of internet applications. In this world, there is a need for tools that can filter information, make suggestions, guide complex tasks or provide other forms of assistance that would reduce the cognitive overhead and workload of the users.

Research in HCI has paid little attention to investigating how such assistive tools can be incorporated into existing user interfaces and developing techniques that combine direct manipulation and adaptive support. Similarly, there is little research on adaptive hypermedia (AH) systems aimed at studying and resolving the usability problems of adaptation.

In this paper, we investigate the above problems and identify gaps in the current literature. We propose a new content adaptation technique (Tsandilas and schraefel 2003a) influenced by fisheye views (Furnas 1986) that permits subtle variations in the adaptive views of hypermedia documents. This technique balances between focus and context by adjusting the size of the visual elements in a document. We discuss the role of focus and context in adaptation techniques and suggest that the use of context can reduce the cost of inaccurate adaptation. We argue that adaptation techniques can be examined and compared in terms of the level of focus and context that they provide. A pilot study comparing fisheye-like adaptation with stretchtext-like adaptation allows us to make preliminary observations about the advantages and disadvantages of these techniques.

In addition to the fisheye-like adaptation technique, we propose an interaction model for AH that merges adaptive support with direct manipulation. The goals of this interaction model are to make the system's adaptive behaviour transparent and predictable and endow the user with rapid and powerful controls over adaptation. The proposed interaction model makes use of the fisheye-like adaptation technique to tightly couple the adaptation process with the interface controllers and support the user's task with continuous and incremental visual feedback. A prototype exhibiting this interaction model was tested by a small group of users. We discuss the user feedback on our model. Finally, we present our overall conclusions and future directions.

2. Problems concerning adaptation

Adaptive systems suffer from three major problems:

1. They depend on the construction of user models that are incomplete and usually erroneous.

2. They result in complex conceptual models that cannot be comprehended by users.
3. They may disable users from having the control of the system's actions.

Although research in the areas of user modelling and machine learning tries to address the first problem by applying new user modelling techniques and new learning algorithms, it is commonly acknowledged that no user model can accurately describe a user. It is also hard to believe that a future intelligent system will be able to predict precisely what users want as even human experts may fail to do so. The second problem derives from the fact that the way that an adaptive system makes decisions and acts may not be clear to users. Adaptive systems build and maintain user models that are hidden from the user. As a result of this, the responses of the system may seem inconsistent and unpredictable. The third problem becomes critical when the system cannot accurately infer the user's needs. User goals may change rapidly. In this case, unless the user gives direct feedback, the system will not have enough evidence to capture any shift in the user's goals.

In addition to the above problems, adaptation in hypertext may affect *landmarks* on which users base their navigational and reading tasks. In general, landmarks are distinctive environmental features functioning as reference points (Vinson 1999). In a document, elements that may act as landmarks are images, textual elements, graphics, structural forms of laying out information, fonts, etc. Users may depend highly on the presence of landmarks when they navigate, so disturbing these landmarks may disrupt their *mental models* and result in disorientation. In the rest of this section, we survey how existing approaches have tried to address the above problems and identify their main limitations.

2.1 Coping with the uncertainty

Recognizing the fact that user goals cannot be predicted with certainty, some approaches formulate the problem of adapting a user interface as a decision-theoretic problem (Zukerman and Albrecht 2001). In this case, the parameters of a user model are estimated with some probability. The decision on which action to be taken by the system is determined by its expected utility in comparison to the expected utility of other possible actions. In general, the expected utility of an action depends on the level of uncertainty in the user model as well as individual costs and rewards of actions taken under known conditions. The most representative work towards this direction is Microsoft's Lumière project (Horvitz *et al*. 1998), which used Bayesian networks to infer user goals and decide on which actions to be taken in assistance to the user. In a more recent work, Horvitz and Apacible (2003) proposed a framework for inferring the cost of interrupting users based on Bayesian models.

The major advantage of the above approaches is that the system's adaptive behaviour is manifested only when its expected utility is large enough; for example, the user needs assistance with a high probability or the cost of interruption is low. However, costs and rewards of adaptive actions depend on several factors and cannot be easily quantified. A still open question is how the selection of an adaptation technique affects the utility of a system's adaptive behaviour. In the area of AH, previous research has tried to classify adaptation techniques (Brusilovsky 2001), but costs or rewards associated with each technique have not been examined.

2.2 Transparency and predictability

Making a system transparent can help its users to build mental models that correctly match the conceptual model of the system. As Hook (2000) observes, transparency is an issue that applies not only to adaptive systems but to other systems as well. She also argues that transparency does not necessarily mean that the system has to explain the internal parts of the system in all their details. People can be good drivers without having a complete model of how the engine of a car works. We can argue that transparency refers mainly to the visibility of the system's runtime behaviour and the context in which this behaviour is demonstrated rather than the visibility of the internal parts of the system. People can drive cars efficiently as long as they have good knowledge of the car's running behaviour and reactions.

On the other hand, an adaptive system's behaviour may vary according to the details of the user model and its inference mechanism, which are usually non-transparent. This is why the actions of an adaptive system seem to be unpredictable. As a solution to this problem, Cook and Kay (1994) suggested that user models should be viewable. As user models can be complex and contain several parameters, the main challenge of this approach is the interpretation of the user model into a form that the user can understand easily. In their system, Cook and Kay provided visualizations of user models, the components of which were organized as interactive hierarchical structures. Different shapes were used to indicate the type of each node in the hierarchy; for example, crosses represented user characteristics and diamonds represented user beliefs. The user could click on the nodes to unfold them and uncover their details. Such views of user models can be complex and hard to assimilate. Furthermore, they are decoupled from the main user interface and not associated directly with the interaction model of the application. Consequently, although the details of such user models are viewable, the process of their construction may not be transparent.

Hook (2000) observes that, depending on the application domain and the individual user's experience, it may be difficult to provide comprehensible views of user models. In this case, it may be appropriate to hide complex inference mechanisms from the user and show, instead, simplified views of the user model that provide a sense of predictability. Several learning systems have used 'skillometers' to give an indication of a student model (Kay 2001). Skillometers enable the learners to see how the system models their progress. Other approaches (Koda and Maes 1996, Horvitz 1999, Andre and Rist 2002) have suggested the use of anthropomorphic agents that imitate human–human communication. These agents are gifted with facial expressions that provide a sort of transparency of what the agent believes about the user's goals. As stated by Shneiderman and Maes (1997), the main argument against anthropomorphic agents is that they give false expectations about their intelligence and their ability to communicate with users.

2.3 Controllability

Researchers usually distinguish between *adaptive* and *adaptable* user interfaces (Fischer 2001). In contrast to adaptive systems, adaptation in adaptable systems is determined mainly by the user and less by the system itself. The main advantage of adaptable systems against adaptive systems is that they give the users control over the process of adaptation and reduce the effect of incorrect system decisions. The cost of the increased controllability is the additional effort required from the user. The user may need to learn the adaptation component before being able to manipulate it.

The distinction between adaptability and adaptivity is usually theoretical as an adaptive system may incorporate adaptable characteristics and allow for some level of user control. User control may have different forms and affect different levels of the system's adaptive behaviour. An empirical study conducted by Jameson and Schwarzkopf (2002) indicated that some users may like to have control over the system's actions, while others may prefer automatic assistance. However, some users may be willing to switch between more or less controllable versions of an interface depending on how their task evolves over time.

Figure 1 exhibits three different types of controllability in adaptive or adaptable systems:

1. Users *customize* the interface by selecting the view that best satisfies their needs or selecting which functionality appears in the interface. The system does not provide any automatic assistance to support this task.
2. Users do not have direct control over the actual interface but instead they control the user model on which the system bases its adaptive behaviour.
3. Users control the level of the system's intrusiveness or the adaptation method.

McGrenere *et al*. (2002) showed that customizable user interfaces may have advantages over interfaces that include adaptive features. However, customizable interfaces often require users to have advanced knowledge of the system; for

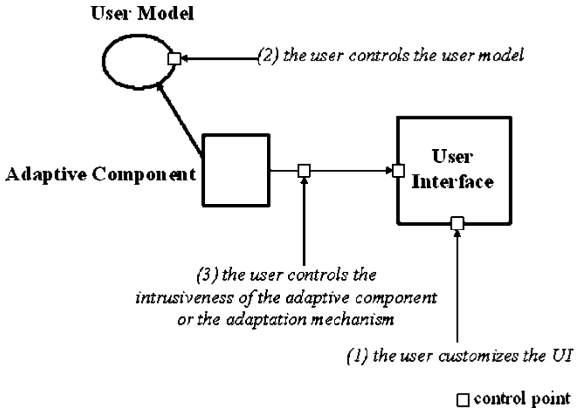

Figure 1. Forms of user control in adaptive/adaptable systems.

example, the user may need to set numerous parameters in a configuration file, while they cannot change dynamically as the needs of the users evolve over time. A recent study on customizable menus (Bunt *et al*. 2004) indicated that novice users may have problems customizing an interface and that adaptive support might assist the customization process.

Controllability of user models has been investigated mainly in the context of intelligent tutoring systems. According to Kay (2001), unlike in early tutoring systems that viewed users as *students*, the term *learner* is now favoured. This implies that the role of users is not passive but they are responsible for their own learning, participating in the construction of their model and the selection of the teaching strategies. Kay has introduced the notion of *scrutable* adaptive systems that enable users to investigate and review the way that the system has been adapted. Tutor is a scrutable AH system that has been developed within this framework (Czarkowski and Kay 2003). At the beginning of each session, Tutor constructs a student model based on the answers of the student to a small set of profile questions. Based on this model, parts of the content may be excluded from a page. At the bottom of each adapted page there is a link to an explanation section. The explanation section explains how adaptation is performed and what content has been excluded from a page. Users can revise their answers to the profile questions by clicking on an icon on the top of the adapted pages. Tutor does not distinguish between user feedback, which is received by the system in order to build the user model, and user control over the system's adaptive behaviour. Both types of interaction are performed by the user by answering a small set of predefined questions. For this reason, this approach cannot be applied to systems in which the construction of user models depends on extensive or implicit user feedback.

Finally, Microsoft's research has tried to tackle the problem of balancing between automated assistance and intrusiveness and investigate how intelligence can be incorporated into direct-manipulation interfaces. Horvitz (1999) refers to this type of interface as a *mixed-initiative* user interface, in which users and intelligent agents collaborate to achieve the user goals. Although such a system's beliefs about the goals of a user are based on implicit user feedback, users are also allowed to explicitly specify utilities and threshold probabilities that affect the system's intrusiveness and adaptation strategy. Users, however, do not get a direct picture of how controlling these parameters affects the behaviour of the intelligent interface. The underlying adaptation model is not transparent to the user and, as a result, the system's behaviour may appear inconsistent and unpredictable.

3. Content adaptation supporting focus and context

Adaptation techniques used by AH systems are usually classified into two main categories: techniques that provide adaptive navigation support and techniques that provide content adaptation (Brusilovsky 2001). Ideally, an adaptation technique should be able to facilitate the task of a user without having negative effects when adaptive actions are not precise. Moreover, an adaptation technique should provide some level of control and transparency and minimize the danger of disrupting the user's mental model by removing landmarks.

In the case of information exploration tasks, an adaptation technique should be able to reduce the information overload and at the same time eliminate the cost of inaccurate guesses of the user's information needs. Based on the previous discussion, we define adaptation as the process of adapting the *focus* of a task while preserving its *context*. The role of context is to (a) help the user to 'contextualize' and therefore better explain the system's adaptive behaviour, (b) preserve distinctive elements that may act as landmarks, and (c) facilitate the switch of the user's current focus. Under this perspective, adaptation techniques can be studied in terms of the level of focus and context that they provide. In this paper, we concentrate on content adaptation.

3.1 Focus, context and fisheye views

Supporting context and focus has been the goal of several techniques in HCI research. Most of these techniques are based on fisheye views (Furnas 1986), which provide both local detail and global context in a single display. Fisheye views have been applied to visualize information in several domains. Furnas (1986) applied fisheye views to program code, tree structures and calendars. Fisheye techniques were used by Sarkar and Brown (1992) to support viewing and browsing graphs. Bederson (2000) applied fisheye zooming to pull-down menus with the goal of reducing the cognitive load caused by long lists of choices. Greenberg *et al.* (1996) introduced fisheye views to support group awareness when multiple people work on the same document.

Techniques based on fisheye views have also been applied to hypertext applications (Noik 1993, Holmquist 1997). These techniques provide fisheye views of collections of web pages or hypertext networks rather than fisheye views of a page's content. Bederson *et al*. (1998), on the other hand, developed the Multi-Scale Markup Language (MSML), a markup language implemented using the HTML <Meta> tag to enable multiple levels of zooming within a single web page. Their goal, however, was to produce interactive web pages that can be zoomed-in and zoomed-out rather than adapt the content of the pages according to user goals or interests.

Fisheye-view techniques define a degree of interest (DOI) function that specifies how the elements of the visualization are presented. The definition of the DOI function is application dependent. Different approaches use different techniques to visualize information with respect to the DOI function. Noik (1993) classifies fisheye-view approaches into two main categories: *filtering* and *distorting* fisheye views. Approaches that belong to the first category use thresholds to constrain the display of information to relevant or interesting elements. Approaches that belong to the second category apply geometrical distortion to the visualization. This is usually performed by altering the positions and the sizes of the visualized elements; for example, elements of interest are zoomed in, whereas irrelevant elements are zoomed out. Fisheye-view techniques usually assume that there is a single focal point, and the value of the DOI function decreases with distance from this point. However, several fisheye approaches (Sarkar and Brown 1992, Greenberg *et al.* 1996) support multiple focal points at the same time.

3.2 Fisheye-like content adaptation

Here we explain how fisheye-view techniques can be applied to content adaptation. Limiting our attention to information exploration tasks, we assume that a user model captures the current interests of the user. In this case, the DOI function is determined by the relevance between the interests of the user and the individual pieces of information. Each page of the hypermedia content is assumed to be segmented into smaller fragments such as sections or paragraphs. If I represents the user interests, the DOI value for a fragment f is:

$$DOI(f) = relevance(f, I) \tag{1}$$

where $relevance(f,I)$ is a measure of similarity between I and f. If f and I are represented by two feature vectors $\vec{f}$ and $\vec{I}$, respectively, the DOI function can be expressed as the cosine similarity between the two vectors (Salton 1991):

$$DOI(f) = \cos(\vec{f}, \vec{I}). \tag{2}$$

According to this definition, the value of DOI for a particular fragment of a page grows as the user's interests become relevant to the content of the fragment. This definition differentiates from the original conception of fisheye views. Proximity

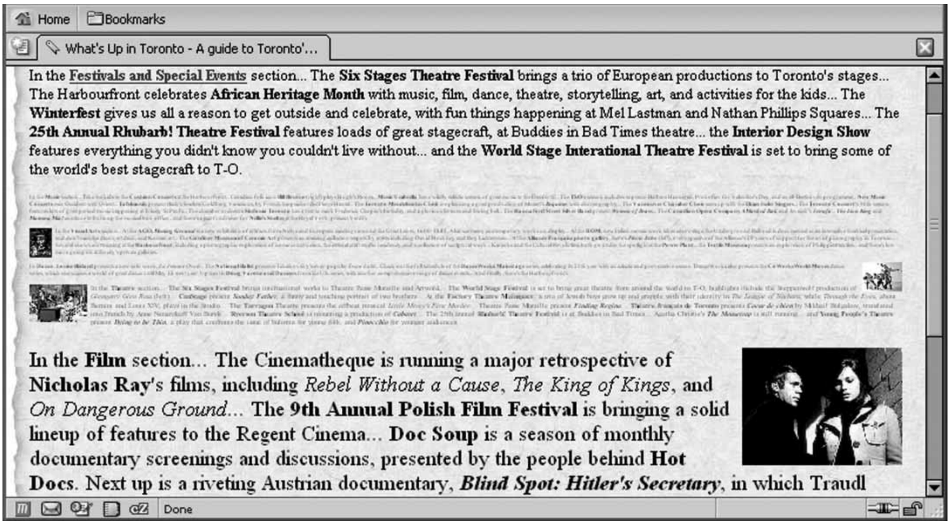

Figure 2. Fisheye view of a web page.

is not measured in terms of geometrical distance, but it refers to the semantic distance between the content of the different segments on the page. Furthermore, the focal point is determined by the focus of the user's interests rather than the user's current focus of attention. As multiple segments on a page may be relevant to the current interests of the user, multiple focal points are supported.

Figure 2 presents a distorted version of a web page where the DOI function determines the size of the visible elements of each paragraph. In the example in figure 2, the user is interested in music events. Therefore, paragraphs that relate to music are shown with larger fonts whereas irrelevant paragraphs are minimized. Image sizes are also adapted with respect to the containing paragraphs. In general, if l_{max} and l_{min} are the maximum and minimum size, respectively, of a visual element within a fragment f, adaptation is achieved by setting the size of the element to $l = \max(l_{max}\, DOI(f), l_{min})$, where the range of values of the DOI function has been normalized between 0 and 1.

An advantage of fisheye views over other visualization techniques is that they preserve landmarks of the information that appears as context. As shown in figure 2, distinct structural elements of the page such as pictures, layout and number of paragraphs are maintained; they are, however, distorted. Two experimental studies conducted by Skopik and Gutwin (2003) on distortion-based fisheye views of graphs revealed that distortion may not disturb the spatial memory of users as long as users can identify and trust landmarks such as distinctive nodes in the visualized space. In addition to the fact that the above adaptation technique preserves features of the page's layout that may act as landmarks, it allows readers to get direct feedback about the quantity and structure of the material within the minimized paragraphs.

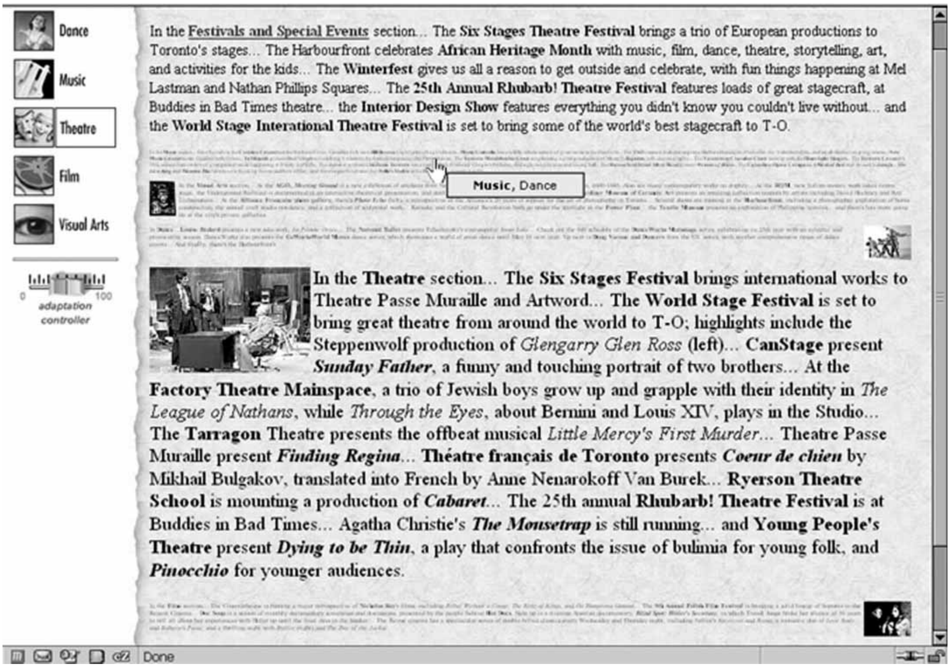

Figure 3. Use of glosses to give feedback about the content of minimized fragments.

3.3 User control

Although the fisheye-like adaptation technique preserves information about fragments that are out of the focus of a user's interests, the actual content of minimized fragments may not be legible. Therefore, users may not be able to examine the subject of a fragment and verify its connection to their goals. To address this problem, we enhanced the fisheye technique with a set of interaction techniques that increase user control over adaptation. Figure 3 shows the use of glosses that provide hints about the content of a paragraph when the user hovers the mouse over the paragraph. In addition to the use of glosses, we incorporated a mechanism that allows fluid transitions of individual paragraphs between context and focus. This mechanism resembles Fluid Links (Zellweger *et al*. 1998). More precisely, by double-clicking on a paragraph that is out of focus, the user can zoom in on the text of the paragraph together with its containing images. Animation is used to change the zooming level smoothly. If the user double-clicks again, the paragraph is zoomed out to its initial size. This mechanism can be considered as a local rather than a global change of focus. The global adaptation of the page is not affected when a paragraph is double-clicked. In other words, temporary changes in the user's attention are not translated into switches of the user's current interests.

The left portion of the page shown in figure 3 contains widgets that give the user additional control over the adaptation process. More specifically, it contains a menu with icons and titles that represent stereotypes of user interests. The menu illustrates the current focus of navigation on which adaptation at the right part of the page is based. It also lets the user change the current focus by selecting a different icon. The left part of the page also contains a slider that adjusts the level of context by setting the minimum size l_{min} of the visual elements. When the value of the slider is zero, adaptation has no effect on the appearance of the pages. However, when the value of the slider is maximum, non-relevant fragments disappear, which means that no context is provided. In other words, the slider allows the user to zoom in (zoom out) to more (less) adaptive versions of a hyperdocument.

3.4 Context and uncertainty

As discussed earlier, the expected utility of an adaptive action is determined by (a) the rewards or costs of taking the action given specific user goals, and (b) the uncertainty about the goals of the user. Focus and context compete with each other by affecting the cost or reward of the adaptive actions. For example, assume that a user is interested in information about theatre. Adapting the pages so that music-only events are highlighted has a cost as the user may be distracted from his or her initial task. If the adaptation technique does not provide any context, for example paragraphs about theatre are hidden, the cost is high, whereas if context is provided the cost may be lower. On the other hand, context implies additional information and provides opportunity for serendipitous exposure to something that may be interesting to the user. If adaptation satisfies the needs of the user, extra non-relevant information causes additional information overload, which may reduce the utility of the adaptation. In conclusion, context should be viewed as an adaptation parameter that can be used to maximize the utility of a system's adaptive behaviour given the uncertainty about the information needs of the user. A major strength of the fisheye adaptation technique is that it enables continuous and smooth transitions between subsequent levels of context. This allows for subtle adjustments of the costs and rewards that are associated with the level of context that adapted pages provide.

3.5 Comparing content adaptation techniques

The fisheye adaptation technique presented above can be considered as a new technique for adapting canned text. Brusilovsky (2001) identifies five techniques for adapting canned text: (a) inserting or removing fragments, (b) altering fragments, (c) stretchtext, (d) sorting fragments, and (e) dimming fragments. Variations of these techniques have been used by different systems in various domains. However, no previous study has tried to examine and compare the techniques. Past evaluations of AH systems have only focused on comparing the adaptive system with its non-adaptive version. We are not aware of any evaluation comparing two or more content adaptation techniques applied to the same system.

3.5.1. Focus and context support in content adaptation techniques. A useful way of studying and evaluating content adaptation techniques is comparing them in terms of the level of focus and context that they provide. Adaptation by removing or altering fragments supports only focus but not context. This means that it may be sensitive to the inaccuracies of a user model. Adaptation by sorting fragments provides both focus and context although the boundaries between focus and context may not be clear. The main disadvantage of reordering the fragments on a page is that it may disturb the natural flow of the information within a page. The techniques that best support context and relate highly to the fisheye-like technique are stretchtext and dimming.

Stretchtext enables users to expand and collapse additional text within a page. MetaDoc (Boyle and Encarnacion 1994) was the first system that used stretchtext as an adaptation technique. It provided different views of hypertext documents for users with different expertise. PUSH (Hook *et al*. 1996) also used stretchtext to adapt the content of hypertext documents to different information tasks. The advantage of these approaches is that although text that is judged as irrelevant or redundant is hidden, the user can open it by clicking on a *hot-word*, which can be text or a representative icon. The amount of context that is provided by this approach depends on the ability of the hot-word to inform the user about the content of the hidden fragment. Stretchtext adaptation can be viewed as a filtering fisheye technique in which content is hidden when its DOI value is below a certain threshold. Compared to the distorting technique, the main disadvantages of stretchtext are: (a) it does not provide any feedback about the quantity and layout of the hidden information; (b) support of context depends on the selection of a representative text or icon for the adaptable fragment, which is a procedure that needs special design considerations from the author of the hypertext content; and (c) it can visualize only two states of adaptation for each fragment, that is fragments are either visible or hidden.

Rich context is supported by the dimming approach (Hothi *et al*. 2000). Fragments containing information that is out of the user's focus are shaded instead of being hidden or zoomed-out. Information in context, in this case, is rich and directly accessible. However, accessing information that appears in either focus or context involves additional scrolling in comparison to the other adaptation techniques, as the amount of information in the adapted pages is not reduced.

3.5.2. Pilot study and observations. As a first step in getting feedback about the fisheye-like adaptation technique, we conducted a preliminary experiment (Tsandilas and schraefel 2003a) comparing it with stretchtext adaptation. To simplify the evaluation procedure and avoid biased conclusions in favour of one technique rather than the other, we tried to eliminate the differences between the implementations of the two techniques. Therefore, we focused on a single variation of the two techniques, which is the way that out-of-focus paragraphs are visualized. In the case of the fisheye adaptation technique, we used a single level of zooming to present paragraphs in context. The fonts were selected to be legible. The stretchtext version was based on the same implementation. The font size of out-of-focus paragraphs was set to zero. However, each paragraph had a representative title or introductory sentence whose font size was never zoomed-out. The interaction model was the same for both techniques. Users could double-click on the body of the minimized paragraph or the paragraph's title to zoom in or expand, respectively, the paragraph. In a similar fashion, users could minimize or collapse the paragraph. Animation was used in both cases to smooth these transitions. Figure 4 shows two versions of the same page, which correspond to the two techniques that we tested.

Six subjects participated in the study. Subjects had to complete six information locating tasks and six information gathering tasks for each of the two techniques on three different pages. The pages showed information about cultural events in Toronto. The first page contained six paragraphs, the second page contained eight paragraphs, and the third page contained approximately seventy paragraphs. The two smaller pages contained images in addition to text. Subjects were asked to locate or gather information from paragraphs that were either in focus or in context. The main goal of the study was to examine how the two techniques performed in both these cases. Performance was measured in terms of the time that subjects spent to complete each task. We also logged the number of double-click actions. At the end of their session, subjects were asked to rate the two techniques and write down their comments.

Although the small number of subjects did not allow us to derive significant results and make general claims, the pilot study revealed some interesting issues. The results did not show any clear advantage of either of the two techniques in terms of task-completion times. It seems, however, that there was an interaction effect between the size of the pages and the technique used. The fisheye adaptation technique performed better than the stretchtext technique in the case of the two smaller pages. However, the stretchtext technique outperformed the fisheye technique in the case of the large page. This advantage was clearer for tasks that involved out-of-focus information. This outcome can be justified by the fact that the stretchtext pages were much smaller than the fisheye pages. This implies that stretchtext adaptation involves less scrolling and searching time. This issue becomes significant when the adapted document is relatively large. In this case, users have to scan multiple screens before discovering a specific piece of information. As one subject observed, the text of the hot-words in the stretchtext pages provided a concise summary of the content of the hidden paragraphs. Consequently, users did not have to read the actual text of the paragraph in order

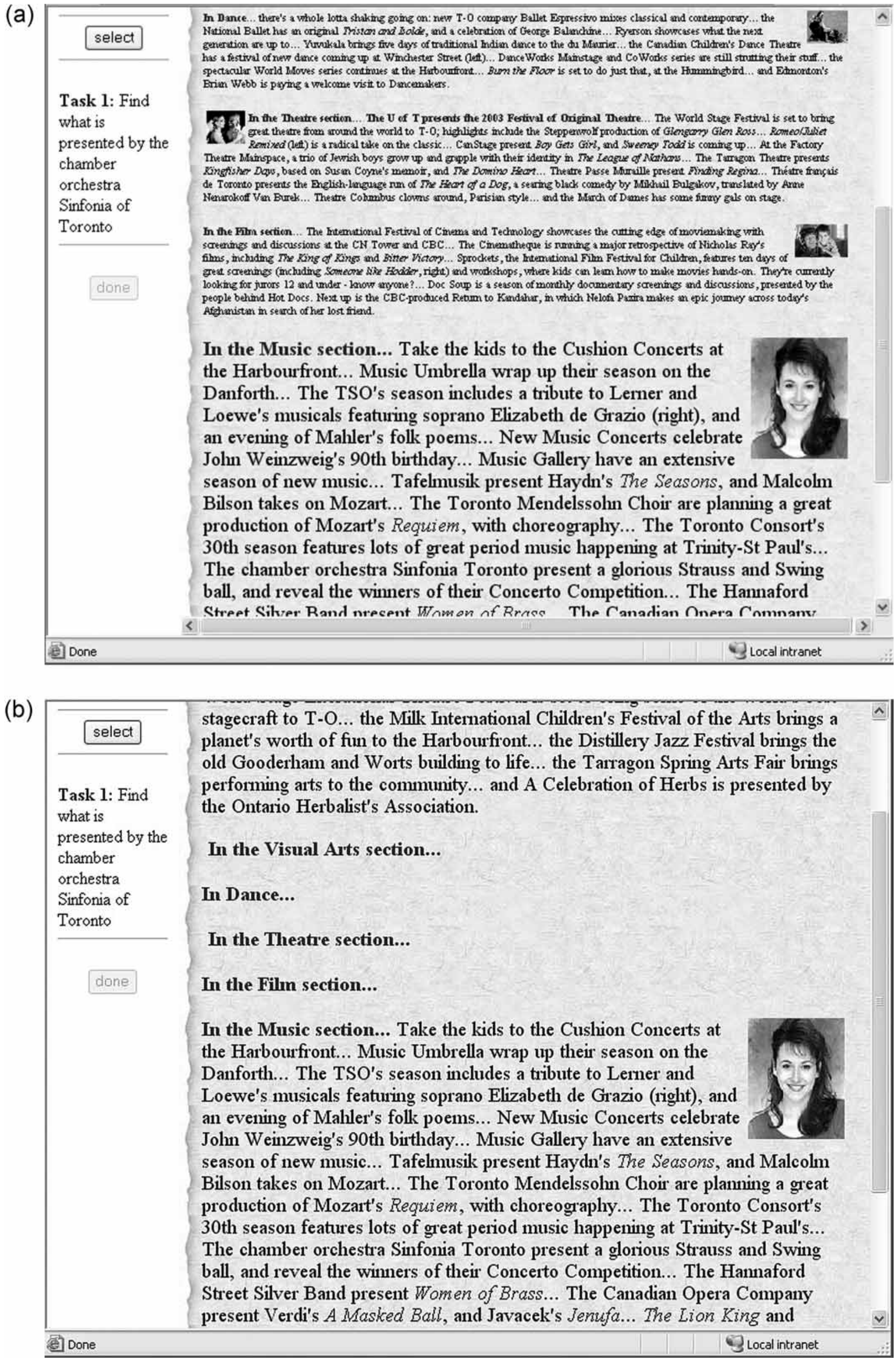

Figure 4. A page adapted by the two tested techniques: (a) distorting fisheye adaptation with two levels of zooming; (b) stretchtext.

to decide on its relevance to their task. On the other hand, four subjects gave a higher overall score to the fisheye technique as they felt that it provided richer information about the content of the out-of-focus paragraphs. This was also manifested by the number of double-click actions, which was clearly lower in the case of the fisheye pages. Subjects often read the content of the minimized paragraphs without zooming in. This practice, however, may have delayed the reading process. The experiment did not measure the cost that is associated with reading small font sizes. Some subjects noted that reading the small fonts required additional effort. Sometimes, they had to move closer to the monitor to read the text.

3.5.3. Conclusions. The pilot study indicated that it is not easy to judge whether one content adaptation technique is better than another. Variables such as the size of the adapted documents, the font sizes, the selection of the hot-words, and the accuracy of adaptation may influence the performance of each technique differently. In addition, the role of context is to provide rich information, which needs, however, to be concise. As the amount of information increases, the problem of information overload becomes greater. In this case, summarization is required. On the other hand, the way the tasks of the experiment were designed did not exhibit the advantage of the fisheye technique in supporting additional context about the layout of the out-of-focus information and preserving landmarks.

4. Transparent and user-controlled adaptation

This section describes the integration of the fisheye-like adaptation technique into an adaptive system. The main objectives of the system's design were (a) to make the user model and the adaptation mechanism transparent, and (b) to support usable controls over the system's adaptive behaviour.

In general, an AH system builds a user model that captures the information needs of a user and then adapts the hypermedia pages with respect to this model. Instead of building global user models, several AH systems (Hirashima *et al.* 1998, Rhodes 2000, Bauer and Leake 2001, El-Beltagy *et al.* 2001) try to capture the local 'context' of navigation. In the simplest case, this context is simply determined by the content of the page that the user currently views (Rhodes 2000). More sophisticated approaches (Hirashima *et al.* 1998, Bauer and Leake 2001) capture the whole navigation history when estimating the underlying context. We adopt a similar but more subtle adaptation approach, where the user model captures the context of the user's interaction with individual page fragments. This approach is tied to the fisheye-like adaptation technique. User feedback is provided in the form of double-click actions on minimized paragraphs. Such feedback allows the adaptive system to reassess the user model and readjust the zooming level of the visual elements on the displayed page.

4.1 User model

Content-based recommendation systems often represent user interests as vectors of terms, also known as *feature vectors*. Each element in a feature vector is weighted according to its relevance to the user's interests. Here, we assume that the vector of user interests $\vec{u}_t$ at time t can be expressed as a linear combination of the vectors of a set of $|I|$ stereotyped user interests:

$$\vec{u}_t = \sum_{i=1}^{|I|} w_{t,i} \vec{v}_i \tag{3}$$

where $\vec{v}_i$ is the vector of the ith stereotype, and $w_{t,i}$ is a weight that shows how relevant to the interests of the user this stereotype is. As illustrated in the following subsections, this assumption allows for the representation of the user model in terms of elements that can be easily visualized, comprehended and controlled by users. The weights in equation (3) are assumed to be normalized so that:

$$\sum_{i=1}^{|I|} w_{t,i} = 1. \tag{4}$$

4.2 Adaptation mechanism

As mentioned earlier, the user model is updated whenever the user performs a double-click on a paragraph. Such an action implies a local shift in the user's interests. Therefore, updating the user model aims at capturing this change. More specifically, if the user double-clicks on the jth paragraph of a page and this paragraph is represented by a vector $\vec{p}_j$, the user model is updated as follows:

$$\vec{u}_{t+1} = (1-a) \cdot \vec{p}_j + a \cdot \vec{u}_t \tag{5}$$

where a is a constant between 0 and 1, which discounts the contribution of the interaction history to the calculation of the user model. Assume now that the vector of the paragraph is expressed as a linear combination of the vectors that represent the stereotypes of user interests:

$$\vec{p}_j = \sum_{i=1}^{|I|} w_{ji} \cdot \vec{v}_i \tag{6}$$

where the weights w_{ji} are normalized to sum up to 1. Then, by combining equations (3), (5) and (6), we can evaluate the new user model as follows:

$$\vec{u}_{t+1} = \sum_{i=1}^{|I|} ((1-a) \cdot w_{ji} + a \cdot w_{t,i}) \cdot \vec{v}_i. \tag{7}$$

When $a = 1$, the zooming actions do not affect the user model, which means that no automatic adaptation is performed. However, when $a = 0$, the history has no effect on the calculation of the user model. In this case, adaptation is based entirely on the content of the manipulated paragraph. As the user zooms in on a paragraph, the page is adapted so that other paragraphs with related content are displayed with large fonts, whereas irrelevant paragraphs are shown with small fonts. In other words, the local focus of the user's interaction coincides with the global focus of adaptation. Finally, in the general case, when the value of a is between 0 and 1, the user model changes progressively while the user interacts with the paragraphs on the adapted pages.

The sizes of the elements on a page are adapted based on the DOI function as defined in equation (2). Animation is used to make transitions between subsequent views of a page fluid and natural. In order to make the adaptation process clear, the animation is performed in two steps. In the first step, the paragraph that is clicked on by the user is zoomed-in. In the second step, the size of the other paragraphs is updated based on the DOI function. This animation effect allows users to distinguish between transitions that reflect changes in the local focus of navigation and transitions that reflect changes in the user model.

4.3 Visualizing the user model

As discussed earlier, the system's adaptive behaviour may be the outcome of several user actions, and the way that these actions can be interpreted by the system is not unique. As a result of this, users may not be able to anticipate the current state of their interaction with the system and understand the adaptation result. The problem can be solved by making the user model transparent. Figure 5 shows two different views of the same page corresponding to two different instances of the user model. Each instance of the user model is visualized on the left frame of the page. The visualization of the user model is based on adjusting the font size of a small set of labels. Each label describes a different stereotype of user interests. Font sizes vary between a minimum and a maximum value that is proportional to the weight of the corresponding vector in the user model. For instance, the user model that defines the first view of the page in figure 5 has weights 0.5 for music, 0.3 for festivals and 0.2 for dance. The other weights are equal to 0. Similarly, the user model that corresponds to the second view has weights 0.5 for music and 0.5 for film. Any change in the user model is reflected immediately to the size of the labels on the left frame of the page. Again, animation is used to smooth transitions between subsequent changes in the font sizes of the labels.

We should note that the above approach makes the system's adaptation behaviour transparent without revealing the actual adaptation mechanism. The user may not know the details about how the system translates his or her actions to infer the user model and how this user model is used to adapt the content of a

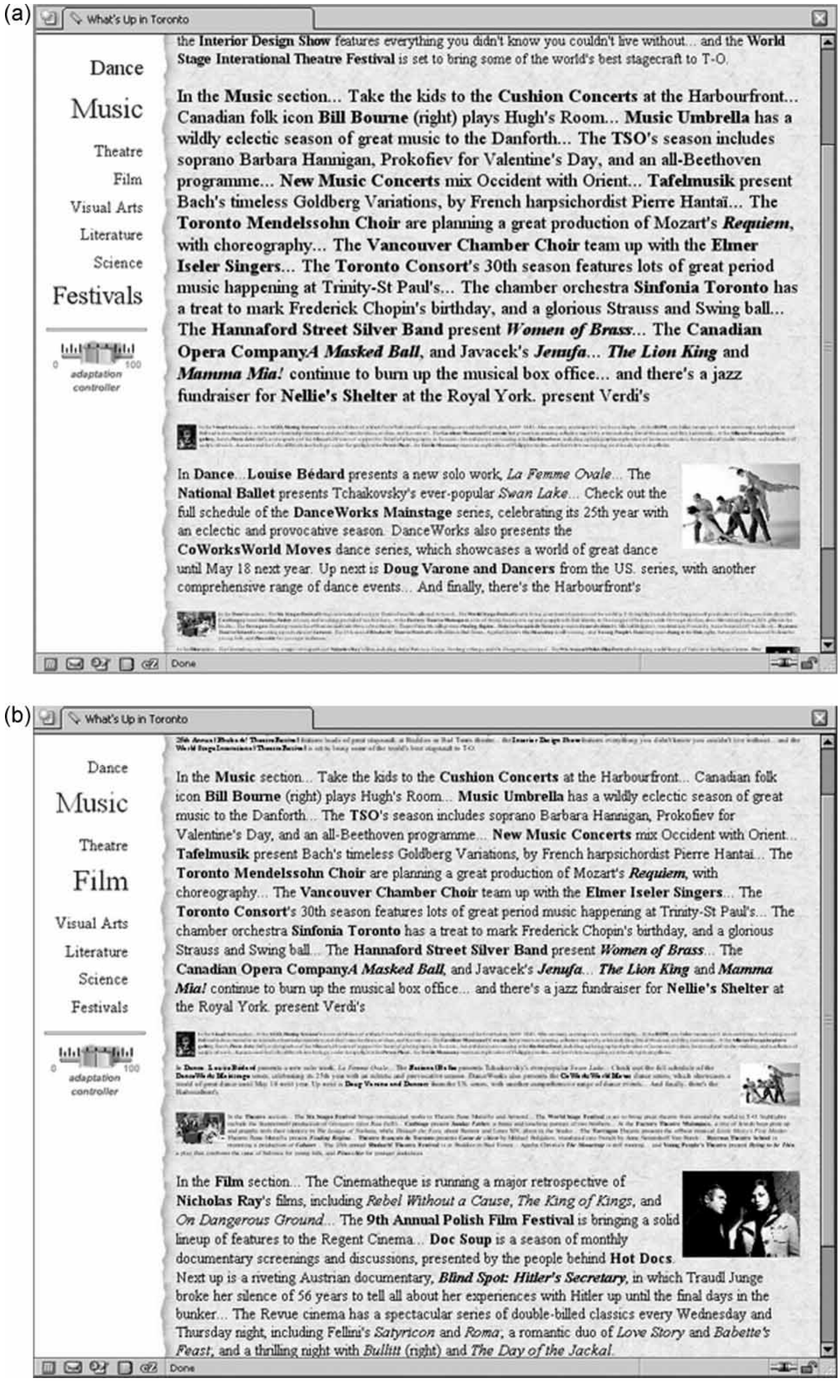

Figure 5. The same page under two different instances of the user model.

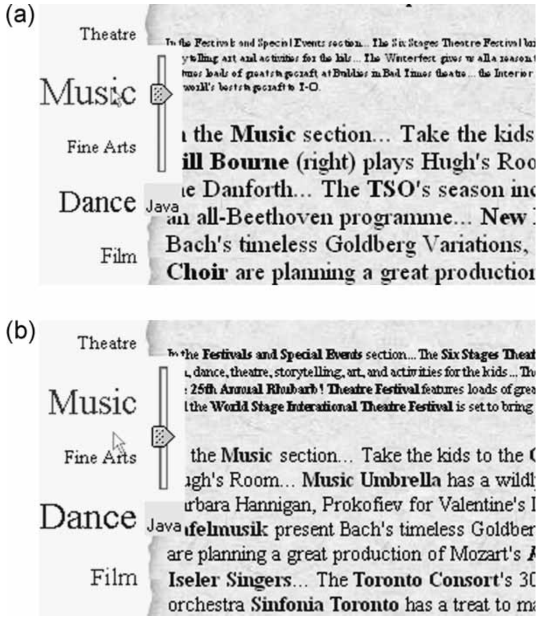

Figure 6. Controlling the user model using a pop-up slider.

page. However, at every moment, the user is aware of the system's state as the user model is always visible. The user receives direct feedback about the effect of every single interaction. Thus, as long as the inference and adaptation mechanisms are reasonable, that is paragraphs are associated with the correct stereotypes of user interests, the user can understand the adaptation mechanism and predict the outcome of their actions.

4.4 Direct manipulation of content adaptation

In previous work (Tsandilas and schraefel 2003) we have demonstrated how link adaptation can be manipulated directly by the user. According to this approach, the user controls the adaptation by handling a set of sliders that correspond to topics of user interests. As the user manipulates a slider, continuous visual feedback is provided that helps the user to comprehend the underlying adaptation mechanism. As the fisheye-like content adaptation technique supports continuous transitions between the adaptive views of a page, a similar approach can be applied to support a direct-manipulation interaction model for content adaptation.

Figure 6 demonstrates how, in our prototype, a user can adjust the weights of the individual vectors of the user model. The user can click on any label on the left frame of the page. This action causes a slider to pop up, which allows the user to adjust the weight of the corresponding vector. The user moves the slider by dragging the mouse. The slider disappears when the mouse is released. This type of slider has been influenced by FaST Sliders (McGuffin *et al.* 2002), which are sliders that pop up when the user performs a rapid gesture over a visual object. They are used to control continuous parameters of the associated object. The use of pop-up sliders eliminates the need to preserve continuously visible controllers whose presence is redundant in a regular interaction mode. The activation of a pop-up slider is fast and requires minimal screen space. Therefore, switching between the adaptive and the adaptable mode of interaction does not require users to shift their attention, for example, to a different window and does not disrupt their main task.

As the value of a slider changes, the size of the label also changes to reflect the updated user model. As the weights in equation (3) are normalized, moving a slider affects all the weights in the user model. When the weight w that a slider controls changes to Δw, then each other weight w_i is updated as follows:

$$w_i' = w_i - \Delta w \cdot \frac{w_i}{\sum_{w_k \neq w} w_k}, \sum_{w_k \neq w} w_k \neq 0 \qquad \text{or} \qquad w_i' = \frac{\Delta w}{|I| - 1}, \sum_{w_k \neq w} w_k = 0. \tag{8}$$

When a slider moves, the size of the paragraph on the page also changes to reflect the new weights of the user model. In this way, users receive continuous feedback about how the content of a page is associated with the stereotypes of user interests.

4.5 Implementation details

The main functionality of the prototype was implemented in JavaScript. A Java applet implements the visualization of the user model and the pop-up sliders. We used LiveConect technology to realize the communication between the Java applet and the JavaScript functions. The vectors that describe the content of the pages were created manually. We should note, however, that automatic generation of such vectors would be feasible by applying simple information-retrieval and text classification techniques (Tsandilas and schraefel 2003b).

5. User feedback

To get a better understanding of the strengths and weaknesses of the proposed interaction model, we conducted an informal user study. Six users participated in the study. All six participants used the web on a daily basis. Four participants were graduate students in computer science, while the others did not have any background in computer science. Each user was shown several versions of the system. Different versions included different combinations of navigational aids; that is, pop-up glosses, visualizations of the user model and pop-up sliders.

To encourage users to interact with the prototypes, they were asked to freely navigate within pages or complete tasks that involved locating specific information. Examples of specific instructions were: ‘locate a reference to a Jazz event’ and 'find events that you would like to attend'. While performing the tasks, each person was asked to explain the system's reactions and justify the adaptation result. The tester did not give any details about why and how the adaptation was performed. Each user spent about 20 to 30 minutes interacting with the different versions of the system. At the end of their session, participants were asked to complete a questionnaire evaluating the use of glosses, the visualization of the user model, the pop-up sliders, and the animation mechanism. The value of the history parameter a shown in equation (5) was set to 0. This made the adaptation mechanism less complex.

5.1 Observations

All the participants exhibited difficulty in trying to describe the adaptation mechanism when both the glosses and the visualization of the user model were disabled. They seemed to understand that paragraphs followed a semantic relationship and that the zooming behaviour somehow respected this relationship. However, most users failed to describe clearly these relationships and explain the system's reactions. On the other hand, the existence of the zooming labels in the user model's visualization helped users to explain the relationships among the paragraphs of the given pages. Furthermore, users were able to describe roughly the meaning of the changing label sizes and characterize the current view of a page. A couple of users, however, felt that even when the zooming labels were displayed, they could not completely understand the adaptation mechanism. The main reason is that some paragraphs seemed to be irrelevant to the categories suggested by the zooming labels, although a more careful reading of the text would reveal that the paragraphs were in fact relevant. An interesting observation is that users may try to invent complex mechanisms to explain accurately the behaviour of a system rather than trying to understand a simple mechanism that does not match precisely their mental model. The two non-computer science participants seemed to be more willing to conform to simple models while ignoring minor inconsistencies.

When available, the pop-up sliders were heavily used by five out of the six users. This group of users seemed to prefer uncovering hidden information by using the sliders rather than double-clicking the minimized paragraphs. One user, however, preferred interacting directly with the paragraphs and identifying the context of the minimized text by using the glosses. Some users found the sliders unintuitive as they kept clicking on the labels rather than dragging them. This was not surprising as users were not accustomed to this type of interaction.

5.2 User answers and comments

All the users agreed that glosses were useful and helped them complete the given tasks. Five users stated that both the zooming labels and the sliders were valuable and helped them to understand the system's behaviour. The same group of users evaluated the complete version of the system, that is the version that included all the navigational aids, with the highest mark. On the contrary, one user reported that the zooming labels and the sliders did not add any value to the user interface. This was the user who interacted directly with the content of the pages and did not use the pop-up sliders. As he explained, observing the left frame of the page distracted him from his main task. He suggested that the state of interaction should be shown close to the area of the current focus. Another user suggested that information currently displayed by glosses and zooming labels should be always visible above each paragraph.

Half of the users reported that the pop-up sliders were easy to use, and half reported the opposite. The negative answers are consistent with the observations that we mentioned above. In addition, two users suggested that the values of the sliders as well as the sizes of the labels should change independently rather than being normalized. However, participants were not given the opportunity to try both versions and compare them. Normalized sliders suggest a more complex conceptual model but minimize the number of actions required to change the weights in the user model.

Finally, all the users liked the animation used to smooth the transitions between the views of the adaptive pages. They all agreed that animation helped them to understand these transitions.

6. Conclusions and future directions

This paper has surveyed usability problems of AH systems. In response to these problems, we have proposed a content adaptation technique and an interaction model aimed at minimizing the cost of adaptive behaviour and increasing user control over adaptation. Based on distorting fisheye views, the proposed adaptation technique provides multiple and continuous adaptive views of hyperdocuments. It also supports smooth transitions between focus and context. As suggested above, by balancing between focus and context, adaptation techniques can decrease the costs of inaccurate decisions made by adaptive systems. The role of context is also important in preserving distinctive visual elements within the adapted page that act as landmarks. We have argued that adaptation techniques can be studied and compared in terms of the amount of focus and context that they provide. The pilot study that we conducted indicates that the usefulness of the context that a content adaptation technique provides may depend on several factors such as the size of the adapted pages. In future work, we plan to investigate the role of context in adaptation and its connection to information overload and the accuracy of adaptation in a more formal setting.

Making the user model transparent and manipulated directly by the user is the main goal of the interaction model suggested by this paper. According to this model, transparency is achieved by providing direct and continuous visual feedback that informs the user about the system's runtime state. The model also provides quick mechanisms for controlling the adaptation process. Control is achieved by manipulating pop-up sliders, whose effect on adaptation is incremental and visualized continuously. In contrast to other approaches (Cook and Kay 1994, Hook *et al.* 1996, Horvitz 1999, Czarkowski and Kay 2003) in which transparency and controllers are separated from the main user interface and mode of interaction, our approach tightly couples the adaptive and adaptable parts of the user interface. Users can interact with hypertext pages in two complementary forms: (a) by reading and manipulating fragments of information within each page, and (b) by manipulating the elements of the user model visualized on the left part of each page. The cost of switching between these two forms of interaction is minimal as it only requires simple clicks of the mouse button without affecting the user's working view. We are considering other types of controllers such as pie and tracking menus (Fitzmaurice *et al.* 2003, Tinz), which could reduce the above cost even more. Such control widgets would allow users to control various parameters of the adaptation by quick gestures without moving their focus to a different part of the adapted page.

Our work assumes that the content of the pages on which adaptation is performed is not totally homogeneous, that is it can be viewed under multiple perspectives. The content of the pages that we used in our prototype was divided into fragments. The content of each fragment was expressed in terms of several stereotypes of user interests such as music, dance and theatre. This may not be feasible in the case of hypertexts that consist of many small pages. Nevertheless, we argue that our approach suggests a new paradigm of authoring and reading hypertexts. Instead of splitting information into small pieces shown on separate pages, several interconnected pieces can be integrated into one larger page. In this way, navigation between different nodes is substituted by moving the focus of adaptation within the fisheye version of the integrated page. Thematic links that usually appear as menus in framed pages are replaced by the visualization and controllers of the user model. This approach enables the use of different classification schemes over the same content, which can be read under multiple perspectives. It also preserves the surrounding context of the information that the user reads or explores and reveals semantic associations among the individual content segments. Applying our approach to larger sets of pages and other domains will help us to further evaluate its usefulness.

Acknowledgements

We thank the participants of the two user studies reported in this paper. We also thank the anonymous reviewers and Bowen Hui for valuable comments on this document. Finally, we thank Luke Murphy for his permission to use the content of 'What's up in Toronto', available online at http://www.whatsuptoronto.com.

References

E. Andre and T. Rist, "From adaptive hypertext to personalized web companions", *Communications of the ACM*, 45, pp. 43–46, 2002.

T. Bauer and D.B. Leake, "Wordsieve: a method for real-time context extraction", in *3rd International and Interdisciplinary Conference, Context*, Dundee, Scotland: Springer Verlag, 2001, pp. 30–44.

B.B. Bederson, "Fisheye menus", in *13th Annual ACM Symposium on User Interface Software and Technology, UIST*, San Diego, CA: ACM Press, 2000, pp. 217–225.

B.B. Bederson, J.D. Hollan, J. Stewart, D. Rogers, A. Druin and D. Vick, "A zooming web browser", in *Human Factors and Web Development*, NJ: Erlbaum, 1998, pp. 255–266.

C. Boyle and A.O. Encarnacion, "MetaDoc: an adaptive hypertext reading system", *User Modeling and User-adapted Interaction*, 4, pp. 1–19, 1994.

P. Brusilovsky, "Adaptive hypermedia", *Modeling and User-adapted Interaction*, 11, pp. 87–110, 2001.

A. Bunt, C. Conati and J. McGrenere, "What role can adaptive support play in an adaptable system?", in *ACM International Conference Intelligent User Interfaces*, Island of Madeira, Portugal, 2004, pp. 117–124.

R. Cook and J. Kay, "The justified user model: a viewable, explained user model", in *4th International Conference on User Modeling*, Hyannis, MA, 1994, pp. 145–150.

M. Czarkowski and J. Kay, "How to give the user a sense of control over the personalization of AH", in *Workshop on Adaptive Hypermedia and Adaptive Web-based Systems, AH2003, User Modeling Session*, 2003, pp. 121–132.

S.R. El-Beltagy, W. Hall, D.D. Roure and L. Carr, "Linking in context", in *12th ACM Conference on Hypertext and Hypermedia*, Aarhus, Denmark: ACM Press, 2001, pp. 151–160.

G. Fischer, "User modeling in human–computer interaction", *User Modeling and User-adapted Interaction*, 11, pp. 65–86, 2001.

G. Fitzmaurice, A. Khan, B. Buxton and G. Kurtenbach, "Tracking menus", in *16th Annual ACM Symposium on User Interface Software and Technology, UIST*, Vancouver, Canada, 2003, pp. 71–79.

G.W. Furnas, "Generalized fisheye views", in *CHI '86*, Boston, MA: ACM Press, 1986, pp. 16–23.

S. Greenberg, C. Gutwin and A. Cockburn, "Using distortion-oriented displays to support workspace awareness", in *People and Computers XI (Proceedings of HCI'96)*, Springer-Verlag, 1996, pp. 299–314.

T. Hirashima, N. Matsuda, T. Nomoto and J. Toyoda, "Context-sensitive filtering for browsing in hypertext", in *3rd International Conference on Intelligent User Interfaces*, San Francisco, CA: ACM Press, 1998, pp. 119–126.

L.E. Holmquist, "Focus+context visualization with flip zooming and the zoom browser", in *Proceedings of ACM CHI '97 Conference on Human Factors in Computing Systems—Extended Abstracts*, ACM Press, 1997, pp. 263–264.

K. Hook, "Steps to take before intelligent user interfaces become real", *Interacting with Computers*, 12, pp. 409–426, 2000.

K. Hook, J. Karlgren, A. Waern, N. Dahlback, C.G. Jansson, K. Karlgren and B. Lemaire, "A glass box approach to adaptive hypermedia", *User Modeling and User-adapted Interaction*, 6, pp. 157–184, 1996.

E. Horvitz, "Principles of mixed-initiative user interfaces", in *ACM SIGCHI Conference on Human Factors in Computing Systems*, Pittsburgh, PA, 1999, pp. 159–166.

E. Horvitz and J. Apacible, "Learning and reasoning about interruption", in *ICMI 2003, ACM International Conference on Multimodal Interfaces*, Vancouver, Canada, 2003, pp. 20–27.

E. Horvitz, J. Breese, D. Heckerman, D. Hovel and K. Rommelse, "The Lumière project: Bayesian user modeling for inferring the goals and needs of software users", in *14th Conference on Uncertainty in Artificial Intelligence*, Madison, WI, 1998, pp. 256–265.

J.K. Hothi, W. Hall and T. Sly, "A study comparing the use of shaded text and adaptive navigational support in adaptive hypermedia", in *International Conference of Adaptive Hypermedia and Adaptive Web-based Systems, AH 2000*, Trento, Italy, 2000, pp. 335–342.

A. Jameson and E. Schwarzkopf, "Pros and cons of controllability: an empirical study", in *Adaptive Hypermedia and Adaptive Web-based Systems. Second International Conference, AH2002*, Malaga, Spain: Springer, 2002, pp. 193–202.

J. Kay, "Learner control", *User Modeling and User-adapted Interaction*, 11, pp. 111–127, 2001.

T. Koda and P. Maes, "Agents with faces: the effects of personification of agents", in *HCI'96*, London, UK, 1996, pp. 98–103.

J. McGrenere, R.M. Baecker and K.S. Booth, "An evaluation of a multiple interface design solution for bloated software", in *ACM CHI 2002 Conference on Human Factors in Computing Systems*, ACM CHI Letters, Minneapolis, MN, 2002, pp. 163–170.

M. McGuffin, N. Burtnyk and G. Kurtenbach, "FaST sliders: integrating marking menus and the adjustment of continuous values", in *Graphics Interface*, Calgary, Alberta, Canada, 2002, pp. 25–34.

E.G. Noik, "Exploring large hyperdocuments: fisheye views of nested networks", in *Proceedings of the 5th ACM Conference on Hypertext*, Seattle, Washington: ACM Press, 1993, pp. 192–205.

B.J. Rhodes, "Margin notes: building a contextually aware associative memory", in *4th International Conference on Intelligent User Interfaces*, New Orleans, LA: ACM Press, 2000, pp. 219–224.

G. Salton, "Developments in automatic text retrieval", *Science*, 253, pp. 974–979, 1991.

M. Sarkar and M.H. Brown, "Graphical fisheye views of graphs", in *CHI '92 Conference*, Monterey, CA: ACM Press, 1992, pp. 83–91.

B. Shneiderman and P. Maes, "Direct manipulation vs. interface agents", *Interactions, ACM*, 4, pp. 42–61, 1997.

A. Skopik and C. Gutwin, "Finding things in fisheyes: memorability in distorted spaces", in *Conference on Graphics Interface GI'03*, Halifax, Canada, 2003. Available online at: http://www.graphicsinterface.org/proceedings/2003/217/.

J. Tinz, "Radialcontext". Available online at: http://www.radialthinking.de/radialcontext/.

T. Tsandilas and m.c. schraefel, "Adaptive presentation supporting focus and context", in *Workshop on Adaptive Hypermedia and Adaptive Web-based Systems, AH2003*, Nottingham, UK, 2003a, pp. 193–204.

T. Tsandilas and m.c. schraefel, "User-controlled link adaptation", in *the 14th ACM Conference on Hypertext and Hypermedia, HT'03* ACM Press, Nottingham, UK, 2003b, pp. 152–160.

N.G. Vinson, "Design guidelines for landmarks to support navigation in virtual environments", in *CHI '99*, Pittsburgh, PA: ACM Press, 1999, pp. 278–285.

P.T. Zellweger, B.-W. Chang and J.D. Mackinlay, "Fluid links for informed and incremental link transitions", in *Proceedings of the 9th ACM Conference on Hypertext and Hypermedia: Links, Objects, Time and Space–Structure in Hypermedia Systems*, Pittsburgh, PA: ACM Press, 1998, pp. 50–57.

I. Zukerman and D.W. Albrecht, "Predictive statistical models for user modeling", *User Modeling and User-adapted Interaction*, 11, pp. 5–18, 2001.