RWTH Aachen

Department of Computer Science
Technical Report

Scrolling Through Time: Improving Interfaces for Searching and Navigating Continuous Audio Timelines

Eric Lee, Henning Kiel and Jan Borchers

ISSN 0935–3232 · Aachener Informatik Berichte · AIB-2006-17

RWTH Aachen · Department of Computer Science · December 2006

Scrolling Through Time: Improving Interfaces for Searching and Navigating Continuous Audio Timelines

Eric Lee, Henning Kiel and Jan Borchers

Lehrstuhl für Informatik X
RWTH Aachen University, Germany
Email: {eric, kiel, borchers}@cs.rwth-aachen.de

Abstract. Existing work has produced a variety of techniques to improve interfaces for navigating an audio timeline. These interfaces typically map user input to either a change in play *rate*, or playback *position*. Audio feedback while scrolling at arbitrary rates can be provided by: *skipping* immediately to the new position in the audio; *resampling* the audio, which introduces pitch-shifts; *time-stretching* the audio to preserve the pitch; or not at all. We conducted a series of user studies to examine the effects of input and feedback type on targeting performance. Position control was found to be, on average, 15–19% faster than rate control when searching for targets 90 to 100 seconds away in the audio timeline. *Time-stretching* was found to be the best choice in most scenarios, but *skipping*, and, for specific user groups, *resampling*, should be used for precise targeting tasks where the audio play rate falls below one-tenth nominal speed.

1 Introduction

Technology advancements in recent years have made continuous time-based media such as audio and video a popular medium for electronic communication. Increasing availability of computers for content creation, the internet for content distribution, and portable media players such as the iPod (www.apple.com), have dramatically increased both creation and consumption of these digital media types. Radio, for example, has in recent years been reborn in digital form as "podcasts"; with today's technology, it is possible for even the average home user with a computer to create such content, and distribute it to a global audience. According to Apple, over one million podcasts were already subscribed to, just two days after the iTunes podcast directory became available [App05].

With this increased popularity of the medium, searching and navigating through an audio timeline is becoming both more common and more important. Audio navigation tasks can also be roughly categorized as *searching* or *gisting* (obtaining the essence of a message). Searching differs from gisting in that searching typically requires at least a passing familiarity with the material; searching also often requires the user to locate a target with some level of accuracy. Common scenarios for searching include:

- finding a specific section in a podcast listened to yesterday
- looking for the start of the next track in a live concert recording that has no track marks
- skipping over a commercial in a podcast
- extracting and editing segments from a recorded voiceover for a CHI 2007 video submission

Unfortunately, most commonly-available interfaces for navigating the timeline of audio do not differ significantly from the "tape recorder" metaphors of *play*, *stop*, *fast forward* and *rewind* from the 1950's. Recent versions of QuickTime Player (www.quicktime.com) and Windows Media Player (www.microsoft.com) support audio playback at variable rates, but is controlled using an "advanced" setting that is neither accessible nor visible in the main user interface.

Existing systems often make use of audio processing techniques to assist with audio searching, including automatic speech recognition (ASR) [WHA+02], speaker segmentation [DMS92], and emphasis detection [Aro97]. Unfortunately, these algorithms are unable to assist with finding detailed semantic information in an audio stream [SHWD04]; thus, in the general case, one must still listen to the stream to find the desired information.

There is, however, little work that studies the various methods to interpret user input, or provide feedback in an audio navigation interface. Research in document navigation, in contrast, has examined in detail both rate controls (where the user controls the scrolling speed) and position controls (where the user controls the scrolling position) [HCBM02][ZSS97], and both input types are used in practice. Existing work in improving audio navigation interfaces [Aro97][HLBG05], however, favors rate control, often with the implicit assumption that they are superior to position control.

The temporal nature of audio creates unique challenges when designing feedback for navigation interfaces – naïvely altering the play rate of audio by resampling it creates well-known pitch-shifting artifacts (commonly known as "the chipmunk effect"). Despite the increasing prevalence of time-stretching algorithms that perform extra processing to remove the pitch shift, systems such as audio editing software continue to employ resampling for interactively adjusting the play rate.

In this paper, we present the results of an empirical study examining the effects of input and feedback type on tasks that require the user to navigate an audio timeline to search for a specific target. We begin with a comparison of spatial (document) navigation and temporal (audio) navigation. We then propose a design space for audio navigation interfaces, and show how existing work in this area fits into this space. We then present the results of several user studies that explore this design space in more detail, leading to a set of guidelines for designing audio navigation interfaces.

2 Spatial vs. Temporal Navigation

Navigating through the timeline of continuous time-based media, such as an audio recording, is similar to navigating through a document in many respects. Despite the spatial nature of documents, as opposed the temporal one of audio, the *input* techniques used for scrolling through a document often apply to audio as well. Audio navigation differs from document navigation, however, in how *feedback* is provided to the user while scrolling.

2.1 Input

A common software interface widget for scrolling through audio is the timeline slider, analogous to a scrollbar in a document window. The *wiper* inside the scrollbar, which controls the current viewing area in a document, is the *playhead* in an audio timeline slider. The arrow buttons at either ends of a scrollbar correspond to the *fast forward* and *rewind* buttons (see Fig. 1).

Fig. 1. The scrollbar for document navigation (left) is analogous to the timeline slider for audio navigation (right).

Zhai et al. [ZSS97] observed, however, that the scrollbar interface for navigating through a document suffers from at least three drawbacks: time is required to acquire the wiper; scrollbars are ill-suited for continuous scrolling with precision; and navigating to the scrollbar shifts the user's locus of attention away from the target. The timeline slider uses the same mappings, and thus suffers from similar drawbacks; moreover, the playhead in a timeline slider is usually very small, and thus acquiring it is even more difficult than with a scrollbar (see Fig. 1).

As a result, alternative interfaces to more efficiently navigate a document have been studied. These interfaces typically control *rate* (user input maps to scrolling velocity), or *position* (input maps to viewing area position). For audio navigation, a rate control would similarly control the play rate, and a position control the current playhead position.

Unlike spatial navigation, where both position and rate controls have been studied extensively and their merits widely debated [Zha95], rate controls appear to be the de facto standard for audio navigation; the sliders for playback speed in QuickTime Player and Windows Media Player, and even the ubiquitous *fast-forward* and *rewind* are rate controls.

2.2 Feedback

While spatial and temporal navigation may share many similarities in input, they differ in how feedback is presented to the user. Hürst observes that when navigating through continuous time-based media, only the smallest unit (e.g., one video frame, or one audio sample) can be conveyed to the user at any moment in time [HS02]. In spatial media such as a text document, many lines of text can be displayed to the user at the same time. Moreover, temporal media must often be perceived over time; a single audio sample, for example, has no meaning by itself.

Fortunately, workarounds to this limitation have been developed. Audio editors, for example, represent the temporal dimension spatially by displaying a section of the audio waveform, thus allowing the user to visualize multiple instants of the audio timeline concurrently. While such visualizations are useful for locating "meta-features", such as pauses between words and sentences, even experienced audio editors are unable to derive the words of a speech recording, or the melody of a song by simply looking at its waveform. Thus, Hürst's statement still holds true, and audio must be interpreted over time to fully understand all nuances of its semantics.

Presenting audio recordings at arbitrary rates can be challenging, however, and a surprising number of audio systems today do not support variable-speed audio playback. Time-stretching algorithms, which preserve the original pitch when the play rate is changed, remain an active area of research [KLB06][LD99][Röb03][VR93].

3 Related Work

Document navigation has been studied more extensively than audio navigation. We provide an overview of both, focusing on the aspects most relevant to our work here.

3.1 Document Navigation

Studies comparing rate and position controls have been conducted since the 1970's: a detailed overview of these works is given in [Zha95]. In [ZSS97], Zhai et al. examine a variety of input devices for scrolling and pointing in documents. They found that an isometric joystick (rate control) outperformed the mouse scroll wheel (position control) for locating a hyperlink when scrolling though a ten-page document. More recently, Hinckley et al. found that scrolling performance with rate and position controls exhibit a crossover effect; while rate control is better for long searches, position control is better for short searches [HCBM02]. Moreover, they found that position control can be significantly improved by using an acceleration function.

In a study comparing a touchpad scroll ring, touchpad scroll zone, and the mouse scroll wheel (all position controls) [Whe03], Wherry found that the scroll ring was faster, with lower error rates, and was most preferred by the users for a spatial navigation task. The circular scrolling metaphor has also been extended to mouse and pen interfaces [MH04][Ss04].

There do not appear to be, however, any existing attempts to generalize the conclusions drawn from the above works to audio navigation.

3.2 Audio Navigation

In [HS02], Hürst proposes several software widgets for scrolling through continuous time-based media; these widgets are controlled using a mouse, and include both position and rate controls. He found rate control using a slider to be the most efficient. Input devices other than the mouse are not included in his study, and it appears to be the only one comparing position and rate control for temporal navigation.

Arons' *SpeechSkimmer* [Aro97] and Hürst et al.'s *Elastic Audio Slider* [HLBG05] are rate controls for skimming through speech. Both use a time-stretching algorithm to play audio at variable rates without pitch-shifting artifacts. The *SpeechSkimmer* improves skimming performance by automatically analyzing the content to reduce or eliminate pauses in the audio; pitch is also analyzed to help the user locate potentially interesting sections in the audio. Alternative input and feedback types are not explored.

DiMaß, our technique for audio navigation using direct manipulation, uses position control. DiMaß also uses a pitch-preserving time-stretching algorithm. There is no comparison of scrolling performance with other input devices.

Moving the playhead back and forth over a waveform visualization in audio editors is known as "audio scrubbing", and is often used to mark cut and trim points in the audio. Many audio editors, such as Audacity (audacity.sourceforge.net), do not provide any audio feedback while scrubbing. Some audio editors, such as Adobe Audition (www.adobe.com) advertise "tape-style" scrubbing, where a rate change introduces a corresponding pitch shift. Video editors, such as Final Cut Pro, use an alternative technique where short snippets of audio are played every time the playhead is moved. Interestingly, there are no well-known audio editors that employ the time-stretching techniques used in [Aro97][HLBG05][LB06].

4 Design Space for Audio Navigation Techniques

The design space we propose is based on our analysis of spatial and audio navigation, and survey of existing audio navigation techniques (see Fig. 2). It consists of two orthogonal axes: input and feedback type. Input types are classified as position (also known as *zero order*, see [Zha95]) or rate (*first order*). Higher order input methods, such as acceleration control (*second order*), have been previously demonstrated to be less efficient compared to zero and first order controls [Pou74], and are thus less common; we leave exploration of higher-order controls for future work.

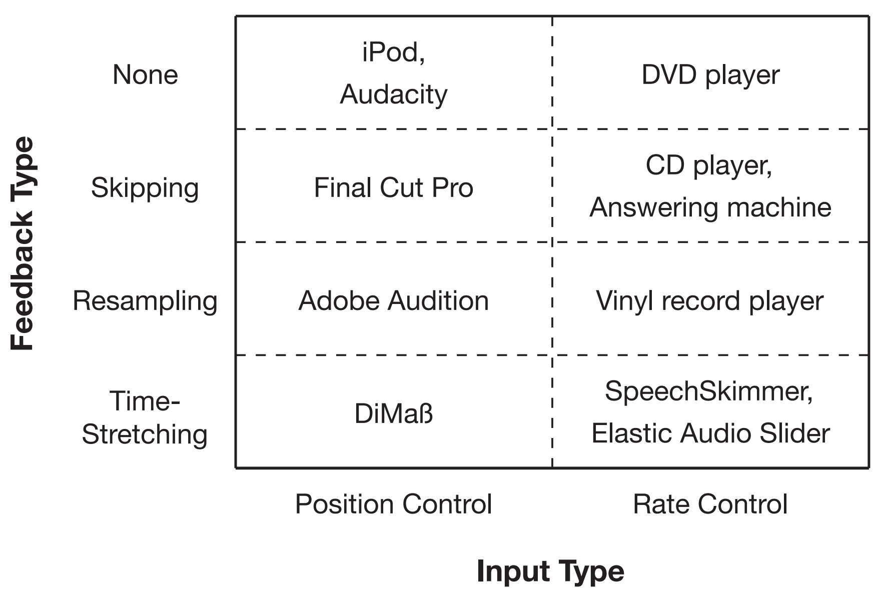

Fig. 2. Design space for audio navigation techniques, populated with examples of existing devices.

We have also identified four possible feedback types[1]:

None: Systems that do not provide audio feedback while scrolling still provide a means to play the audio at its nominal rate (e.g., *play* button). While it may seem obvious that no feedback would result in poor audio scrolling performance, we include it as the baseline case for comparison. It is also common in existing systems – no feedback is given when scrubbing through audio using an iPod or Audacity, for example.

***Skipping*:** A short segment of audio (tens of milliseconds) is played at regular speed when the playhead position is changed. This allows the user to experience feedback at arbitrary scroll rates without any pitch-shifting artifacts. The resulting audio is choppy, however. Many CD players and answering machines provide *skipping* feedback when the *fast-forward* and *rewind* buttons are held down. It is also common in video editors such as Final Cut Pro.

***Resampling*:** The audio is *resampled* to allow playback at arbitrary rates. Resampling also pitch-shifts the audio; the effect is the same as varying the play rate of a vinyl record player. While disc jockeys (DJ's) make use of this feature for artistic effect, pitch shifts to the audio are typically undesirable as it renders the resulting audio incomprehensible. Adobe Audition supports this type of feedback for scrubbing as a separate mode ("tape-style" scrubbing).

***Time-stretching*:** The audio is processed to allow playback at arbitrary rates without changing the pitch. The processing, unfortunately, introduces artifacts into the resulting audio. Time domain algorithms such as waveform similarity overlap-add (WSOLA) [VR93] produce satisfactory results for only a small range of play rates (20% above or below nominal speed); certain audio navigation tasks such as scrubbing, however, easily exceed this range. Moreover, they do not work well for polyphonic audio signals such as music. Frequency domain algorithms, such as the phase vocoder and its variants [KLB06][LD99][Röb03], have been developed to address these limitations, but still exhibit "transient smearing" and "reverberation" artifacts. The *SpeechSkimmer*, *Elastic Audio Slider*, and DiMaß all use time-stretching.

Exploring both of these dimensions simultaneously would not have been practical, and thus we have chosen to study input and feedback type independently.

5 Position vs. Rate control for audio navigation

The following user scenarios illustrate temporal navigation using rate and position controls:

David and Victor are arguing over the name of the ensign killed off in Episode 213 of Star Trek: TOS*, and they decide to refer to the DVD to resolve their dispute. David's DVD player, a Sony DVP-NS700P (www.sony.com), has a shuttle ring control to interactively adjust play rate. Using the DVD menu, David jumps to the scene where the ensign first appears, and uses the shuttle ring to navigate to the part where the his name is mentioned.*

Eva has just returned from a Dave Matthews Band concert. She recorded the entire concert onto her iPod, which the band permits for personal use. Alas, the

[1] We invite readers to view the accompanying video figure for demonstrations of the various input and feedback types.

recording has no track marks, and she must scroll through the audio using the "Click Wheel" to find the start of her favorite song.

Existing research in audio navigation [Aro97][HLBG05] focuses primarily on rate control. We hypothesized that, similar to spatial navigation, position control can be faster for audio navigation under certain circumstances (e.g., when the search target is close to the current position).

We designed an audio navigation experiment where users were asked to locate a target between 90 and 100 seconds from the current playhead position using both rate and position controls. We measured and compared the targeting times for these devices.

5.1 Setup

We included the following three input devices in this study:

Scroll ring: A rate control consisting of a spring-loaded ring. This device is found on certain DVD players, such as David's DVP-NS700P (see Fig. 3a). The play rate increases in the forward direction as it is rotated clockwise, and backward when rotated counter-clockwise. When released, the ring "snaps" back to its original position. Previous research has shown that this self-centering mechanism is an important characteristic for rate control [HCBM02][ZSS97].

Jog dial: A position control using a solid dial. The audio will advance forwards as the dial is rotated clockwise, and backwards when rotated counter-clockwise. Some CD players used by DJ's, such as Stanton's C.500 (www.stantondj.com, see Fig. 3b) offer this type of control.

Touch wheel: The touch wheel operates similarly to the jog dial, except that it is touch-sensitive and thus lacks any haptic feedback. The touch wheel is the primary control on the iPod, where it is used for menu navigation in addition to audio timeline navigation (see Fig. 3c).

(a) (b) (c)

Fig. 3. (a) Scroll ring in the Sony DVP-NS700P DVD player. (b) Jog dial in the Stanton C.500 CD player. (c) Touch wheel on the iPod portable media player.

In our experiments, we used the ShuttleXpress (www.contourdesign.com, see Fig. 4), a device that includes both a scroll ring and a jog dial. Its scroll ring offers 7 unique positions in either direction, which we mapped to play rates from $\pm\frac{1}{4}\times$ to $\pm 16\times$ on an exponential scale. Audio playback stops when the ring is in its rest position. The ShuttleXpress jog dial snaps lightly into place on 10 unique positions per revolution.

For the touch wheel, we used the Phidgets CircularTouch (www.phidgets.com); it is touch sensitive, with a diameter twice as wide as the ShuttleXpress jog dial. The CircularTouch advertises an ability to report 128 unique positions per revolution; in practice, however, we found that the effective resolution was much lower (roughly 32 positions), due to the uncertainty of a finger making contact with a region on the wheel. We repackaged the CircularTouch to resemble a larger version of the iPod Click Wheel (see Fig. 4).

Fig. 4. The ShuttleXpress (left) has an outer self-centering ring for rate control and an inner dial for position control. The Phidgets CircularTouch (right) has a touch-sensitive surface that we used for position control, repackaged to resemble an iPod Click Wheel.

We wrote a software tool that accepts input from all three devices. According to Hinckley et al., incorporating acceleration into a position control significantly improves targeting times [HCBM02]. We used the following acceleration function to compute the play rate r for both the jog dial and touch wheel: $r = (\kappa_s \Delta p/\Delta t)^{\kappa_a}$. Δp is the amount of movement over the time interval Δt, and κ_s and κ_a are constants that control the scroll sensitivity and acceleration, respectively. For this user study, we used values of $\kappa_s = 1.5$ and $\kappa_a = 2$, which we determined in preliminary user tests to give satisfactory results. None of our users reported any problems with the jog dial and touch wheel acceleration.

The user is presented with a slider that shows where the current playhead is in the audio timeline. The playhead position was not reported numerically to the user (see Fig. 5).

In all cases, audio feedback was provided using a high-fidelity, frequency domain time-stretching algorithm based on [KLB06][LD99][Röb03]. For reverse playback, the audio samples were simply played in reverse order; while other methods for reverse playback of speech have been proposed [HLB05], such methods are difficult to apply for non-speech media, and our experiments include both music and speech.

5.2 Procedure

16 volunteers (10 male, 6 female) in their 20's and 30's participated in the experiment; two were office workers, and the remainder were students from varying

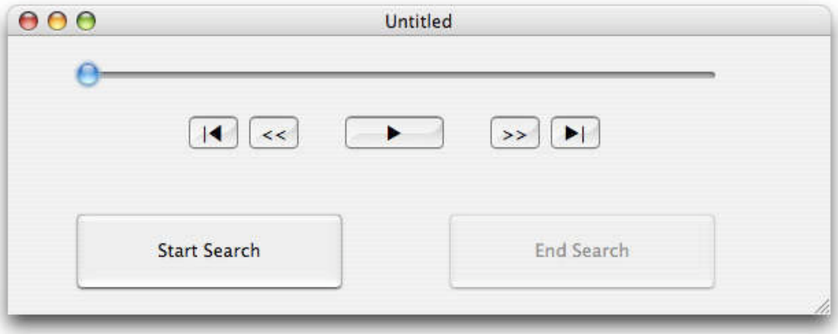

Fig. 5. Software interface presented to users participating in the study comparing position and rate controls.

backgrounds including computer science, psychology, and technical writing. Participants were introduced to the three audio navigation devices, given some time to familiarize themselves with the devices, and then performed some test trials. The actual trials began when they indicated that they felt comfortable operating the devices on their own. The session concluded with an interview and questionnaire; each session took between 30 and 45 minutes. The software was run on a 17" iMac (1.83 GHz Core Duo processor, 1 GB RAM) running Mac OS 10.4.7; Sennheiser HD 280 Pro headphones were used for sound. Participants were compensated for their time with some sweets.

Each trial consisted of locating a cut point in a roughly three minute-long audio recording consisting of either speech or music. Users were instructed to locate the cut point as precisely and quickly as they could. In our subsequent data analysis, we only used data points that were within ± 2 s of the actual cut point. The user was not given any feedback during the session on their accuracy, and we found that in practice, users rarely (less than 10% of the time) were unable to meet this target.

The CircularTouch device gave us intermittent problems, interpreting a finger hovering over the device as input; fortunately, a software restart to trigger a recalibration of the device solved the problem.

Users searching through audio in a realistic situation would know their search target, and to simulate this, we used a relatively easy target: the cut point was either a speaker change (male to female), or a music genre change (classical to pop). Each user was tested with either music or speech, but not both. Users performed two trials for each of the three devices. They were informed beforehand that the cut point would be "somewhere in the middle" of the audio recording. Multiple versions of the same recording with the cut point placed between 90 and 100 seconds into the audio were prepared offline; they were presented to users during the experiment in pseudo-random order. The order in which users operated the devices between trials was also pseudo-randomized – users never operated the same device for two consecutive trials.

5.3 Results and Discussion

The data points for a specific user and device were averaged together, and a repeated measures ANOVA revealed significant differences between the mean search times for the three devices ($F(3, 16) = 5.6$, $p < 0.01$; see Fig. 6). The Tukey HSD post-hoc test revealed that the average search time with the scroll ring is significantly higher than the average search times with both the jog dial and touch wheel. The difference between the jog dial and touch wheel is not significant.

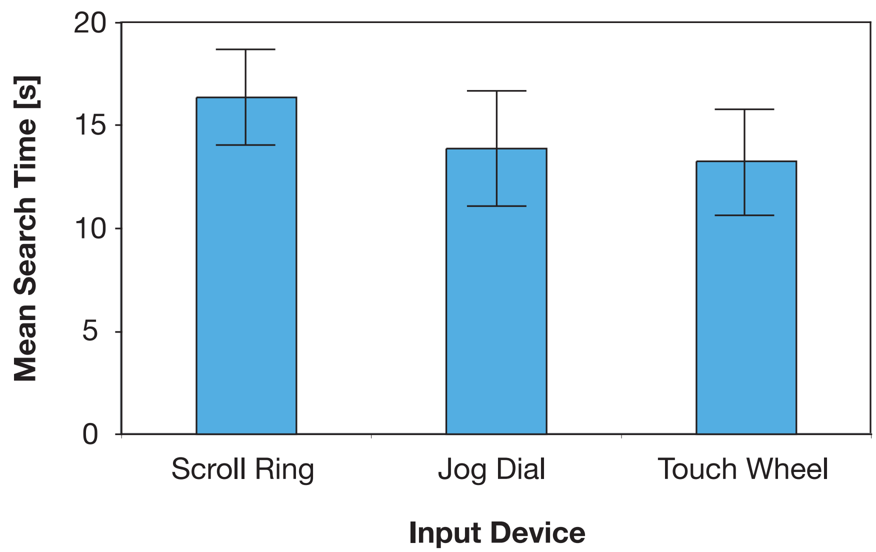

Fig. 6. Mean search times using the scroll ring, jog dial, and touch wheel with 95% confidence bars. The mean search time for the scroll ring is significantly higher than for both the jog dial and touch wheel.

Our results confirm our initial hypothesis that the jog dial and touch wheel, both position controls, allow users to locate a target in a continuous audio stream significantly faster than the scroll ring, a rate control, for search targets between 90 and 100 seconds from the current playhead position. The mean search times are 16.4, 13.9, and 13.3 seconds for the scroll ring, jog dial and touch wheel, respectively, and thus, for our users, the jog dial was 15% faster than the scroll ring, and the touch wheel was 19% faster.

Hinckley showed that for document navigation, position control is faster than rate control for closer search targets, but as the search target distance increases, the performance difference becomes less significant until a "crossover point", when rate control becomes superior [HCBM02]. We believe this same crossover effect applies to audio navigation as well, in which case 90 to 100 seconds is a minimum upper boundary, below which position control is significantly faster than rate control. However, more studies would have to be conducted before such a statement could be conclusively made, which we leave for future work.

Users were also asked to subjectively rate each of the three devices on a scale from 1 ("very poor") to 5 ("very good") (see Fig. 7). Users were also encouraged to provide comments, many of which were consistent with previous studies comparing rate and position controls. Users observed, for example, that it was easier to play through audio at constant rates with the scroll ring, and operating it did not require as much physical movement compared to the dial and wheel. On the other hand, most people felt the dial and wheel were easier to control for precise position changes. Some users had comments specific to the devices we had chosen for our study, such as “the jog dial was too small”, and, “didn’t like the feel/texture of the touch wheel underneath my fingers”.

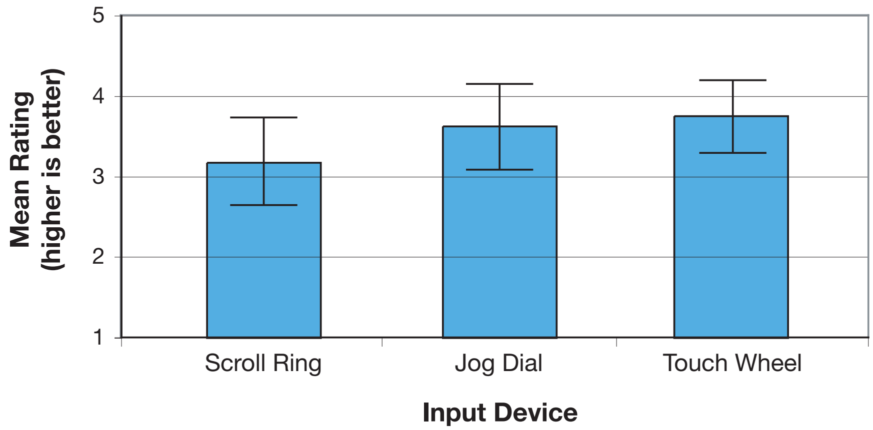

Fig. 7. Mean subjective ratings for the scroll ring, jog dial, and touch wheel with 95% confidence bars (1 = very poor, 2 = poor, 3 = average, 4 = good, 5 = very good).

6 Effect of Feedback Type on Audio Editing

To study the effect of feedback on audio timeline navigation, we chose the following scenario:

Sarah is preparing a two minute summary of an interview for the student radio station. She sits down to extract the most important bits from fifteen minutes of raw material, and remove the “fillers” (the “uhms” and “aahs”). For example, she modifies the phrase “used a.. uhh.. the Heisenberg...” to “used the Heisenberg...”. Unfortunately the software that Sarah is using does not provide any audio feedback when scrubbing over the waveform.

We examined all four feedback types in our design space: no feedback, *skipping*, *resampling*, and *time-stretching*. Both task times and behavioral pattern changes were studied.

We contacted a local student radio station to assist us with our studies. We felt they are representative of an increasing population of hobbyist content-creators: members of the radio station volunteer their time to assemble the radio programs, and come from a variety of backgrounds, including computer science, political science, economics, and geology. Their experience with editing audio ranges from just a few months to a couple of years, and none are professionally trained. Some have a weekly workload of programs to produce, while others contribute only once a month. Interviews make up a significant portion of the time they invest in preparing material to air.

We began by observing how two of these editors use Adobe Audition, the software provided by the radio station, to edit interviews. By default, Audition does not provide any feedback while scrubbing, although the most recent version advertises "tape-style" scrubbing (i.e., *resampling*). However, it must be activated as a separate mode, and we found the resulting lag between cursor movement and playhead position often interferes with normal operation. Moreover, the pitch artifacts introduced by resampling made the resulting audio incomprehensible to the editors. On the other hand, with no audio feedback, the editors were forced to navigate blindly: they would adjust the playhead position, play the audio, readjust the playhead, and so on until they had finally located the target. While being able to visualize the audio waveform is helpful in locating sentence boundaries and pauses, filler sounds are often mixed together with neighboring words, in which case the cut point must still be found using this trial and error method (see Fig. 8).

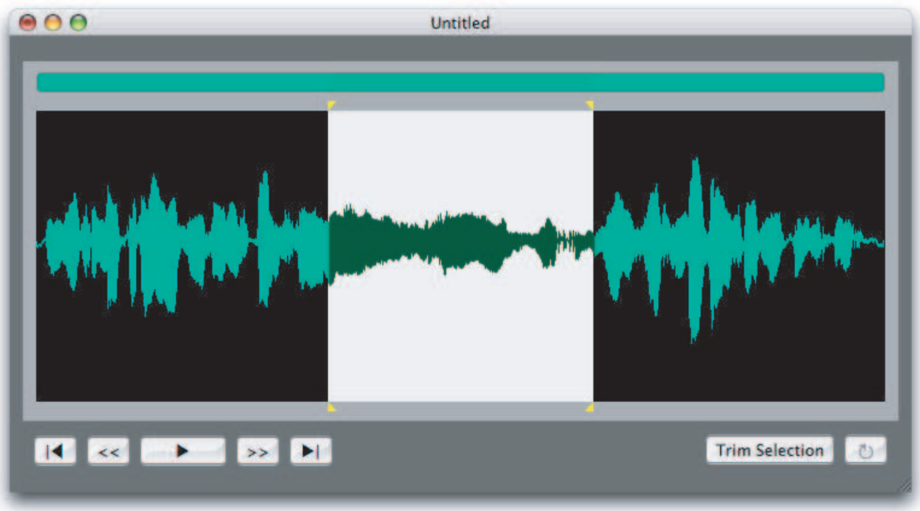

Fig. 8. Example interview segment with a filler. The highlighted region represents the filler to be removed, which cannot be easily identified and correctly marked using only the visual view of the waveform.

We further observed that the following sequence of tasks occur repeatedly while editing an interview:

1. **Locate**: A desired segment, such as a sentence, in the raw recording is located.
2. **Cut**: A "rough cut" of the desired segment is performed, allowing the editor to focus on just that one segment.
3. **Polish**: The rough cut is refined by deleting the extraneous material on either edge of the cut. Fillers, long pauses and mistakes in the speech are also deleted.

The first and third steps were the most time-consuming in the editing process. Note that while a system with automatic content analysis such as the *SpeechSkimmer* could help with the *locate* task, the *polish* task requires precise pinpointing, and the audio must be examined manually.

6.1 Setup

Based on these preliminary observations, we developed an audio editor modeled after the Adobe Audition interface for our user studies (see Fig. 9). The editors were interviewed at depth about the subset of features they used, and these were subsequently incorporated into our software. These features include:

- a waveform view, where most of the interaction occurs
- playhead and selection markers above and below the waveform view; these markers can be dragged to move the current playhead and modify the current selection
- auto-scrolling behavior, with and without selection
- a zoombar located above the waveform, used for scrolling large distances in the audio, as well as interactively adjusting the zoom factor
- shortcut keys for playing, zooming, and deleting the current selection

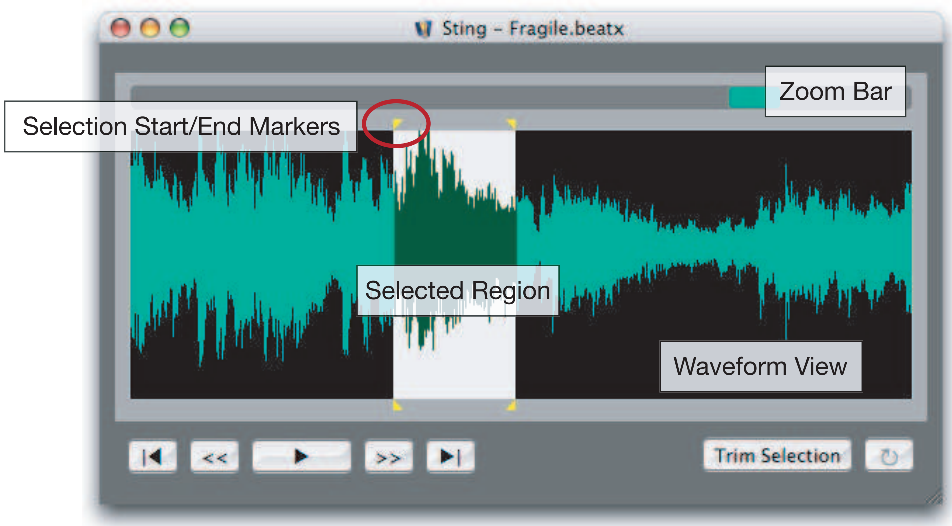

Fig. 9. Audio editing tool modeled after Adobe Audition. The interface consists of a waveform view, where the user can select regions of the audio. The selection start and end points are located above and below the waveform view (the playhead is not visible in this picture). The zoombar above the waveform view is used for scrolling quickly through the audio and interactively adjusting the zoom factor.

We also included a control that allows the user to set the audio feedback type while scrubbing to one of the four feedback types: none, *skipping*, *resampling*, and *time-stretching*. Audio feedback was only given when scrubbing in the main waveform view, or manipulating the playhead and selection markers, but not the zoombar. As the zoombar is used primarily used to change the zoom factor, and for quickly jumping and scrolling to distant points in the audio, we felt audio feedback would not have been appropriate, and this was later confirmed by our users.

Skipping was implemented by playing a 70 ms snippet of audio at the current playhead position whenever it is moved; these snippets are windowed to minimize pops and clicks in the audio. *Resampling* uses the high-quality resampler unit in Core Audio, the audio framework in Mac OS X, and *time-stretching* uses the same algorithm from our previous experiment. As users had previously complained about the lack of responsiveness when testing the tape-style scrubbing feature in Audition, we adapted the synchronization technique presented in [LB06] to minimize the lag between the mouse input and the subsequent playhead movement when providing feedback using *resampling* and *time-stretching*.

6.2 Evaluation With Professional Video Editors

We had the opportunity to demonstrate our prototype to three experienced video editors consulting in UI/workflow design at a company producing video editing software. The editors were familiar with all the feedback techniques except *time-stretching*. After a demo, the editors were given the opportunity to experiment with the prototype using each of the feedback types with both music and speech.

They described their typical workflow as one that is very similar to the *locate/cut/polish* pattern we observed earlier with the radio editors. For cutting audio, all of them preferred feedback provided by *time-stretching*. Surprisingly, however, for precise cuts, two of the editors preferred *resampling*. They revealed that audio editing before computers became common required them to manipulate physical reels of tape, which exhibited the same pitch-shifting effects. And, although it requires training and experience, cutting is actually easier with audio that is lowered in pitch, because certain features, such as transients (drum beats, or a gunshot sound), become more easily recognizable. Thus, to them, the pitch-preserving capability in *time-stretching* did not offer any advantages when doing precise positioning. Moreover, they strongly disliked the transient smearing artifacts and other distortions introduced by the processing at low play rates; *resampling* does not have these artifacts despite the pitch shifts it introduces.

However, they all agreed that editors without prior experience working with tape would most likely not find *resampling* useful.

6.3 Study A: Start-to-Finish Editing

We recruited 9 editors from the student radio station (5 male, 4 female) in their 20's and 30's to participate in this first study. Following the results from our earlier observations with the radio editors and preliminary evaluation with the professional editors, we dropped *resampling* from the experiment. The remaining feedback types were labeled "A", "B", and "C" (no feedback, *skipping*, and *time-stretching*, respectively) when presented to users.

Each session lasted approximately sixty minutes; users were asked to perform complete editing tasks using our software. Each session began with a brief introduction and some test trials. Trials followed the *locate/cut/polish* sequence, and users were asked to indicate when they had completed each task so that their completion times could be recorded. How users went about completing the task was otherwise left to their own best judgment, however. Software was run on a 15" MacBook Pro (1.83 GHz Intel Core Duo processor, 2 GB RAM) running

Mac OS 10.4.7; a Logitech optical mouse and AKG K55 headphones were also used.

Users began with a seven minute interview segment; a text transcription of the recording was given to them on a separate sheet of paper. This transcription simulates the context that our editors would normally have: the person cutting the interview is almost always the person who recorded it. A sentence roughly five seconds long in the audio was highlighted on the transcript, which the users had to locate, extract and polish by removing the fillers and unnecessary pauses. Each user performed six trials, twice for each of three feedback types. The order in which the feedback types were presented were pseudo-randomized, with the feedback type always changing between consecutive trials. Each trial used a completely different recording, although the same six recordings, presented in pseudo-random order, were used for all users. The raw material was graciously loaned to us by the radio station, taken from one of their older archives.

The session concluded with a questionnaire and interview, and users were offered a small amount of monetary compensation for their time.

6.4 Results and Discussion - Study A

Each trial in the Start-To-Finish study took between 105 and 210 seconds to complete, with an average completion time of 159, 133, and 146 seconds with no feedback, *skipping* and *time-stretching*, respectively. Mean completion times divided by task are shown in Fig. 10. As expected, the results indicate a clear trend that *cut* times are reduced when audio feedback is provided, with *time-stretching* giving the best results. There also exists a trend where audio feedback reduces *polish* times. Surprisingly, however, *polish* times were higher with *time-stretching* for three of our users than with no feedback at all, and was almost double that of *skipping* for one user. These results were explained in subsequent interview results.

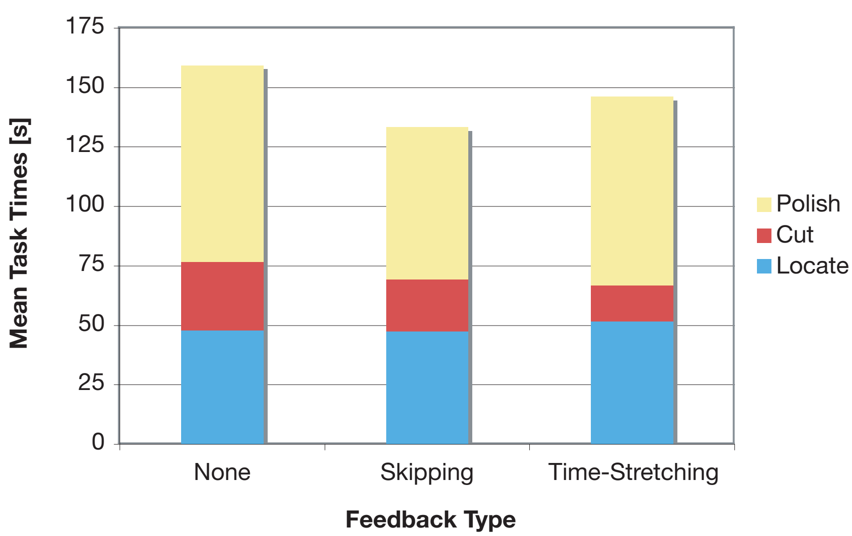

Fig. 10. Mean completion times, divided by task for each feedback type.

The effect of audio feedback on the *locate* task was least dramatic, unsurprisingly, since using the mouse as a position control physically limits the user's ability to navigate large distances [ZSS97].

Users were asked to give a subjective rating of each of the feedback types on a scale from 1 ("very poor") to 5 ("very good", see Fig. 11). Two users ranked *time-stretching* worse than no feedback, and only one user felt *skipping* was slightly worse than no feedback. There appears to be no clear distinction between ratings for *skipping* and *time-stretching*, however. Three of our users were not able to distinguish between *skipping* and *time-stretching*, and rated them identically.

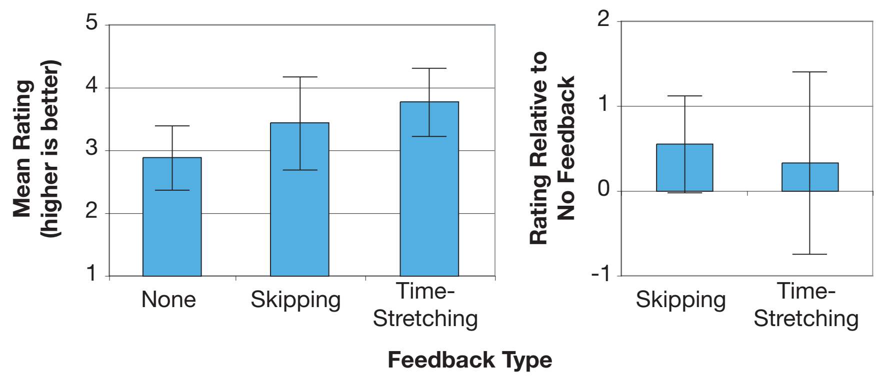

Fig. 11. Mean subjective ratings for the three feedback types (left), and ratings when compared to no feedback (right), with 95% confidence bars. While *skipping* is consistently ranked higher than no feedback, *time-stretching* results are mixed.

Asking users to comment further about their perceived differences between *time-stretching* and *skipping* revealed further results. A majority of the users strongly preferred *time-stretching* during the *cut* task, as expected, since audio processed using *time-stretching* exhibits significantly less artifacts than *skipping*. However, during the *polish* task, more users preferred *skipping*. During the *cut* task, the audio play rate rarely drops below one-quarter nominal speed, but during *polish*, they work at significantly higher zoom rates (two or three seconds of audio, as opposed to tens of seconds during a rough cut, in a window 1280 pixels wide); scrubbing through the audio at such high zoom factors results in audio play rates that are, on average, less than one-tenth nominal speed. Time-stretching at such extreme rates results in significant artifacts, enough for the feedback to be unhelpful, and even objectionable. Users commented that with *skipping*, they were still able to recognizes the syllables in the speech, and could thus more easily complete the task.

These results are consistent with those obtained in our preliminary evaluation with professional editors, although in their case they preferred *resampling*.

6.5 Study B: *Polish*-Only Editing

In this follow-up study, the conditions under which editing was performed were more strictly controlled to help us better understand users' behavior patterns when editing. 7 users from Study A participated in this study, and we recruited an additional 4 editors from the radio station for a total of 11 users (7 male, 4 female). Each trial consisted of removing a single filler from a three second phrase, or five words from a three second segment of music. Users were only able select, play, and delete the current selection. They were not, for example, permitted to change the zoom level of the waveform. Each user was tested with three recordings: two speech and one music. The same recordings were used for each of the three feedback types, but interleaved between trials to minimize learning effects; we hypothesized that any residual learning effects that might occur would be less than the variance introduced by using different data sets. We logged all mouse and keyboard input for offline analysis.

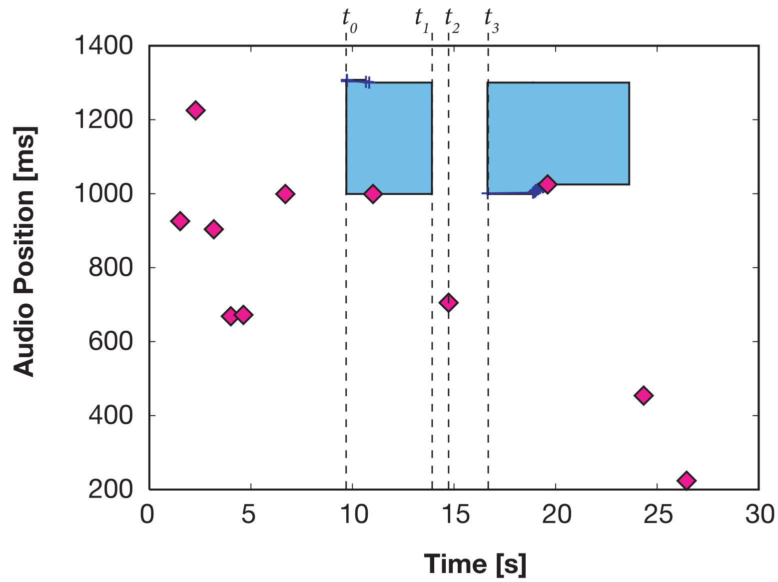

Fig. 12. Removing filler with no feedback. Shaded regions represent selection, diamonds represent times where audio playback is triggered, and crosses are mouse events. The user starts by playing the segment a few times to locate the start of the filler; a region is selected at $t_0 = 9.75$s, and deleted at $t_1 = 13.9$s. The audio is played to verify the cut ($t_2 = 14.7$s), and the user realizes he has removed too much. The deletion is undone at $t_3 = 16.6$ s, adjusted and deleted again. The edited segment is played back twice to verify the edit is now satisfactory.

6.6 Results and Discussion - Study B

Examining the logs captured during the experiment verified our hypothesis that having audio feedback does indeed alter users' behavior while editing. Sample plots of the interaction for the removal of a single filler sound with no feedback and *skipping* are shown in Fig. 12 and 13. These plots show how the audio feedback reduces the uncertainty when users select the filler to cut; users can rely on the audio feedback to help align the selection start and end points, instead of using trial and error.

A quantitative analysis of task times did not reveal any further results beyond those obtained in Study A. This is not surprising, given the open-ended nature of the task we asked users to perform: they would often verify their cut by playing the audio a variable number of times, for example. While it would have been possible to create a setup where all such variability is removed, we felt such a test would not be representative of a real usage scenario, thus bringing into question what the implications of any obtained results would be.

7 Design Implications

Based on the above results, we propose the following guidelines for designing audio navigation interfaces:

For targets close to the current playhead position in the audio timeline, a circular positioning device such as a jog dial or touch wheel is significantly faster

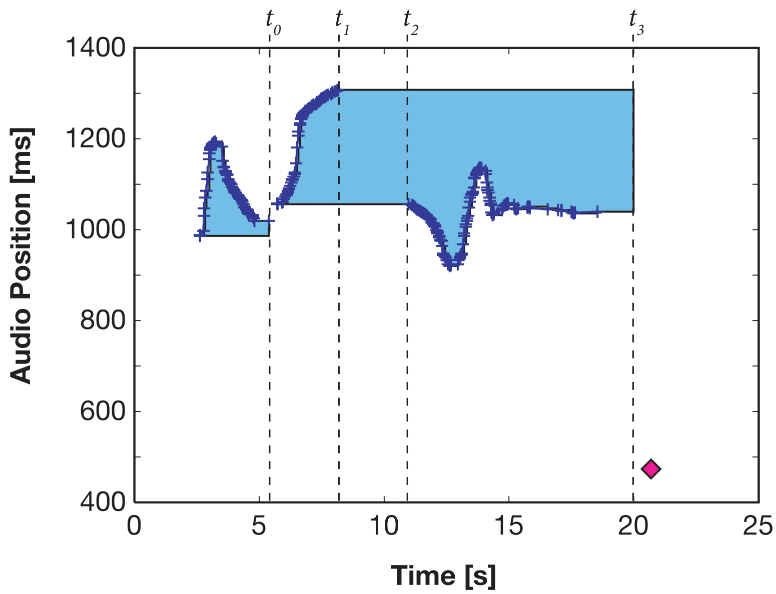

Fig. 13. Removing a filler with *skipping* feedback. After an initial start to locate the starting cut point ($t_0 < 5.5$s), the user selects a region and adjusts the ending cut point until $t_1 = 8.2$ s. At $t_2 = 11$ s, he goes back to fine-tune the starting cut point. He deletes the selection ($t_3 = 20$s) and plays the edited clip once to verify.

than a rate control such as a scroll ring. Based on prior research in document navigation, we believe 90 to 100 seconds is a minimum upper boundary, below which position controls are significantly faster. Users also prefer these position controls over rate control for searching in this range.

Time-stretching should be employed for searching tasks where the play rate of the audio does not drop below one-quarter nominal speed. Not only is audio time-stretched at extremely low rates disturbing to the user because of the artifacts it introduces – targeting performance can become worse than if no audio feedback was provided at all. We believe this threshold value will change as time-stretching algorithms continue to improve, however.

For targeting tasks where the play rate frequently drops below one-tenth nominal speed, either *resampling* or *skipping* should be used for audio feedback. *Resampling* should be utilized for users with prior experience working with tape, as they have the ability to recognize certain cues more easily with audio shifted down in pitch. Most users, however, will prefer *skipping* feedback.

8 Conclusions and Future Work

Interfaces to navigate through an audio timeline can be constructed using a variety of input and feedback types. User input can be used to control the playback position or the play rate; audio feedback at play rates not equal to 1 can be provided via *skipping*, *resampling*, *time-stretching*, or not at all. We performed a series of user studies examining this design space for audio navigation, from which we uncovered some unexpected results. Rate controls are frequently used in audio navigation interfaces; we discovered, however, that position control using the jog dial or touch wheel is, on average, 15–19% faster than rate control with the scroll ring for targets 90 to 100 seconds away in the audio timeline. We hope to conduct further user tests with a larger variety of distances to verify that Hinckley's "crossover effect" for document navigation also applies to audio navigation.

Time-stretched audio produces results closest to what most users expect when audio play rate is altered. If the task requires the audio to be time-stretched at extremely low (less than $\frac{1}{10}\times$) rates, however, feedback provided using *time-stretching* can result in worse targeting performance than no audio feedback. In these situations, we found *skipping* and *resampling* to be two alternatives to assist users for such precise targeting tasks. Designers often automatically disregard *resampling* as a viable method for audio play rate modification, as the pitch shifts it produces render the audio incomprehensible. However, certain user groups, such as audio editors familiar with cutting using reels of tape, will prefer *resampling* over other types of feedback. These results indicate that a hybrid feedback mechanism could be developed for certain applications such as audio editors. *Time-stretching* would be used by default, for example, but when the audio is zoomed in for fine cuts, the system could switch to either *skipping* or *resampling* feedback.

We limited the scope of this work to navigation through an audio timeline, a purely temporal medium. Similar studies could, of course, be performed with other media types, such as video. Video differs from audio in that it contains spatial semantics, in addition to temporal: a time-instant of video can still be interpreted as an image. While we expect the effects of input and feedback type to be similar with such media, we hope to verify them in future studies.

As continuous, time-based media and devices become increasingly ubiquitous, we hope our work will inspire continued research in improved interfaces to navigate audio, and also increase their adoption in popular devices.

References

App05. Apple Computer. iTunes podcast subscriptions top one million in first two days. Press Release, June 2005.

Aro97. Barry Arons. SpeechSkimmer: a system for interactively skimming recorded speech. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 4(1):3–38, 1997.

DMS92. Leo Degen, Richard Mander, and Gitta Salomon. Working with audio: integrating personal tape recorders and desktop computers. In *Proceedings of the CHI 1992 Conference on Human Factors in Computing Systems*, pages 413–418, Monterey, USA, 1992.

HCBM02. Ken Hinckley, Edward Cutrell, Steve Bathiche, and Tim Muss. Quantitative analysis of scrolling techniques. In *Proceedings of the CHI 2002 Conference on Human Factors in Computing Systems*, pages 65–72, Minneapolis, USA, 2002.

HLB05. Wolfgang Hürst, Tobias Lauer, and Cédric Bürfent. Playing speech backwards for classification tasks. In *Proceedings of the ICME 2005 International Conference on Multimedia and Expo*, Amsterdam, The Netherlands, July 2005. IEEE.

HLBG05. Wolfgang Hürst, Tobias Lauer, Cédric Bürfent, and Georg Götz. Forward and backward speech skimming with the elastic audio slider. In *Proceedings of the 19th British HCI Group Annual Conference*, Edinburgh, Scotland, 2005.

HS02. Wolfgang Hürst and Patrick Stiegeler. User interfaces for browsing and navigation of continuous multimedia data. In *Proceedings of NordiCHI 2002*, pages 267–270, Århus, Denmark, 2002.

KLB06. Thorsten Karrer, Eric Lee, and Jan Borchers. PhaVoRIT: A phase vocoder for real-time interactive time-stretching. In *Proceedings of the ICMC 2006 International*

Computer Music Conference, pages 708–715, New Orleans, USA, November 2006. ICMA.

LB06. Eric Lee and Jan Borchers. DiMaß: A technique for audio scrubbing and skimming using direct manipulation. In *Proceedings of AMCMM 2006 Audio and Music Computing for Multimedia Workshop*, Santa Barbara, USA, 2006.

LD99. Jean Laroche and Mark Dolson. Improved phase vocoder time-scale modification of audio. *IEEE Transactions on Speech and Audio Processing*, 7(3):323–332, 1999.

MH04. Tomer Moscovich and John F. Hughes. Navigating documents with the virtual scroll ring. In *Proceedings of the UIST 2004 Symposium on User Interface Software and Technology*, 2004.

Pou74. E. C. Poulton. *Tracking skill and manual control*. Academic Press, New York, 1974.

Röb03. Axel Röbel. Transient detection and preservation in the phase vocoder. In *Proceedings of the ICMC 2003 International Computer Music Conference*, pages 247–250, Singapore, 2003. ICMA.

SHWD04. Risto Sarvas, Erick Herrarte, Anita Wilhelm, and Marc Davis. Metadata creation system for mobile images. In *Proceedings of MobiSys 2004 International Conference on Mobile Systems, Applications, and Services*, pages 36–48, Boston, USA, 2004.

Ss04. Graham Smith and m.c. schraefel. The radial scroll tool: Scrolling support for stylus- or touch-based document navigation. In *Proceedings of the UIST 1994 Symposium on User Interface Software and Technology*, pages 53–56, Santa Fe, USA, 2004.

VR93. Werner Verhelst and Marc Roelands. An overlap-add technique based on waveform similarity (WSOLA) for high quality time-scale modification of speech. In *Proceedings of the ICASSP 1993 International Conference on Acoustics, Speech, and Signal Processing*, volume II, pages 554–557. IEEE, 1993.

WHA+02. Steve Whittaker, Julia Hirschberg, Brian Amento, Litza Stark, Michiel Bacchiani, Philip Isenhour, Larry Stead, Gary Zamchick, and Aaron Rosenberg. SCANMail: a voicemail interface that makes speech browsable, readable and searchable. In *Proceedings of the CHI 2002 Conference on Human Factors in Computing Systems*, pages 275–282, Minneapolis, USA, 2002.

Whe03. Elaine Wherry. Scroll ring performance evaluation. In *Extended Abstracts of the CHI 2003 Conference on Human Factors in Computing Systems*, pages 758–759, Ft. Lauderdale, 2003.

Zha95. Shumin Zhai. *Human Performance in Six Degree of Freedom Input Control*. PhD thesis, University of Toronto, Toronto, Canada, 1995.

ZSS97. Shumin Zhai, Barton A. Smith, and Ted Selker. Improving browsing performance: A study of four input devices for scrolling and pointing tasks. In *Proceedings of INTERACT 1997 Conference on Human-Computer Interaction*, pages 286–292, Sydney, Australia, 1997.

Aachener Informatik-Berichte

This is the list of all technical reports since 1987. To obtain copies of reports please consult
`http://aib.informatik.rwth-aachen.de/` **or send your request to: Informatik-Bibliothek, RWTH Aachen, Ahornstr. 55, 52056 Aachen, Email:** `biblio@informatik.rwth-aachen.de`

1987-01 * Fachgruppe Informatik: Jahresbericht 1986
1987-02 * David de Frutos Escrig, Klaus Indermark: Equivalence Relations of Non-Deterministic Ianov-Schemes
1987-03 * Manfred Nagl: A Software Development Environment based on Graph Technology
1987-04 * Claus Lewerentz, Manfred Nagl, Bernhard Westfechtel: On Integration Mechanisms within a Graph-Based Software Development Environment
1987-05 * Reinhard Rinn: Über Eingabeanomalien bei verschiedenen Inferenzmodellen
1987-06 * Werner Damm, Gert Döhmen: Specifying Distributed Computer Architectures in AADL*
1987-07 * Gregor Engels, Claus Lewerentz, Wilhelm Schäfer: Graph Grammar Engineering: A Software Specification Method
1987-08 * Manfred Nagl: Set Theoretic Approaches to Graph Grammars
1987-09 * Claus Lewerentz, Andreas Schürr: Experiences with a Database System for Software Documents
1987-10 * Herbert Klaeren, Klaus Indermark: A New Implementation Technique for Recursive Function Definitions
1987-11 * Rita Loogen: Design of a Parallel Programmable Graph Reduction Machine with Distributed Memory
1987-12 J. Börstler, U. Möncke, R. Wilhelm: Table compression for tree automata
1988-01 * Gabriele Esser, Johannes Rückert, Frank Wagner Gesellschaftliche Aspekte der Informatik
1988-02 * Peter Martini, Otto Spaniol: Token-Passing in High-Speed Backbone Networks for Campus-Wide Environments
1988-03 * Thomas Welzel: Simulation of a Multiple Token Ring Backbone
1988-04 * Peter Martini: Performance Comparison for HSLAN Media Access Protocols
1988-05 * Peter Martini: Performance Analysis of Multiple Token Rings
1988-06 * Andreas Mann, Johannes Rückert, Otto Spaniol: Datenfunknetze
1988-07 * Andreas Mann, Johannes Rückert: Packet Radio Networks for Data Exchange
1988-08 * Andreas Mann, Johannes Rückert: Concurrent Slot Assignment Protocol for Packet Radio Networks
1988-09 * W. Kremer, F. Reichert, J. Rückert, A. Mann: Entwurf einer Netzwerktopologie für ein Mobilfunknetz zur Unterstützung des öffentlichen Straßenverkehrs
1988-10 * Kai Jakobs: Towards User-Friendly Networking
1988-11 * Kai Jakobs: The Directory - Evolution of a Standard
1988-12 * Kai Jakobs: Directory Services in Distributed Systems - A Survey

1988-13 * Martine Schümmer: RS-511, a Protocol for the Plant Floor
1988-14 * U. Quernheim: Satellite Communication Protocols - A Performance Comparison Considering On-Board Processing
1988-15 * Peter Martini, Otto Spaniol, Thomas Welzel: File Transfer in High Speed Token Ring Networks: Performance Evaluation by Approximate Analysis and Simulation
1988-16 * Fachgruppe Informatik: Jahresbericht 1987
1988-17 * Wolfgang Thomas: Automata on Infinite Objects
1988-18 * Michael Sonnenschein: On Petri Nets and Data Flow Graphs
1988-19 * Heiko Vogler: Functional Distribution of the Contextual Analysis in Block-Structured Programming Languages: A Case Study of Tree Transducers
1988-20 * Thomas Welzel: Einsatz des Simulationswerkzeuges QNAP2 zur Leistungsbewertung von Kommunikationsprotokollen
1988-21 * Th. Janning, C. Lewerentz: Integrated Project Team Management in a Software Development Environment
1988-22 * Joost Engelfriet, Heiko Vogler: Modular Tree Transducers
1988-23 * Wolfgang Thomas: Automata and Quantifier Hierarchies
1988-24 * Uschi Heuter: Generalized Definite Tree Languages
1989-01 * Fachgruppe Informatik: Jahresbericht 1988
1989-02 * G. Esser, J. Rückert, F. Wagner (Hrsg.): Gesellschaftliche Aspekte der Informatik
1989-03 * Heiko Vogler: Bottom-Up Computation of Primitive Recursive Tree Functions
1989-04 * Andy Schürr: Introduction to PROGRESS, an Attribute Graph Grammar Based Specification Language
1989-05 J. Börstler: Reuse and Software Development - Problems, Solutions, and Bibliography (in German)
1989-06 * Kai Jakobs: OSI - An Appropriate Basis for Group Communication?
1989-07 * Kai Jakobs: ISO's Directory Proposal - Evolution, Current Status and Future Problems
1989-08 * Bernhard Westfechtel: Extension of a Graph Storage for Software Documents with Primitives for Undo/Redo and Revision Control
1989-09 * Peter Martini: High Speed Local Area Networks - A Tutorial
1989-10 * P. Davids, Th. Welzel: Performance Analysis of DQDB Based on Simulation
1989-11 * Manfred Nagl (Ed.): Abstracts of Talks presented at the WG '89 15th International Workshop on Graphtheoretic Concepts in Computer Science
1989-12 * Peter Martini: The DQDB Protocol - Is it Playing the Game?
1989-13 * Martine Schümmer: CNC/DNC Communication with MAP
1989-14 * Martine Schümmer: Local Area Networks for Manufactoring Environments with hard Real-Time Requirements
1989-15 * M. Schümmer, Th. Welzel, P. Martini: Integration of Field Bus and MAP Networks - Hierarchical Communication Systems in Production Environments
1989-16 * G. Vossen, K.-U. Witt: SUXESS: Towards a Sound Unification of Extensions of the Relational Data Model

1989-17 * J. Derissen, P. Hruschka, M.v.d. Beeck, Th. Janning, M. Nagl: Integrating Structured Analysis and Information Modelling

1989-18 A. Maassen: Programming with Higher Order Functions

1989-19 * Mario Rodriguez-Artalejo, Heiko Vogler: A Narrowing Machine for Syntax Directed BABEL

1989-20 H. Kuchen, R. Loogen, J.J. Moreno Navarro, M. Rodriguez Artalejo: Graph-based Implementation of a Functional Logic Language

1990-01 * Fachgruppe Informatik: Jahresbericht 1989

1990-02 * Vera Jansen, Andreas Potthoff, Wolfgang Thomas, Udo Wermuth: A Short Guide to the AMORE System (Computing Automata, MOnoids and Regular Expressions)

1990-03 * Jerzy Skurczynski: On Three Hierarchies of Weak SkS Formulas

1990-04 R. Loogen: Stack-based Implementation of Narrowing

1990-05 H. Kuchen, A. Wagener: Comparison of Dynamic Load Balancing Strategies

1990-06 * Kai Jakobs, Frank Reichert: Directory Services for Mobile Communication

1990-07 * Kai Jakobs: What's Beyond the Interface - OSI Networks to Support Cooperative Work

1990-08 * Kai Jakobs: Directory Names and Schema - An Evaluation

1990-09 * Ulrich Quernheim, Dieter Kreuer: Das CCITT - Signalisierungssystem Nr. 7 auf Satellitenstrecken; Simulation der Zeichengabestrecke

1990-11 H. Kuchen, R. Loogen, J.J. Moreno Navarro, M. Rodriguez Artalejo: Lazy Narrowing in a Graph Machine

1990-12 * Kai Jakobs, Josef Kaltwasser, Frank Reichert, Otto Spaniol: Der Computer fährt mit

1990-13 * Rudolf Mathar, Andreas Mann: Analyzing a Distributed Slot Assignment Protocol by Markov Chains

1990-14 A. Maassen: Compilerentwicklung in Miranda - ein Praktikum in funktionaler Programmierung (written in german)

1990-15 * Manfred Nagl, Andreas Schürr: A Specification Environment for Graph Grammars

1990-16 A. Schürr: PROGRESS: A VHL-Language Based on Graph Grammars

1990-17 * Marita Möller: Ein Ebenenmodell wissensbasierter Konsultationen - Unterstützung für Wissensakquisition und Erklärungsfähigkeit

1990-18 * Eric Kowalewski: Entwurf und Interpretation einer Sprache zur Beschreibung von Konsultationsphasen in Expertensystemen

1990-20 Y. Ortega Mallen, D. de Frutos Escrig: A Complete Proof System for Timed Observations

1990-21 * Manfred Nagl: Modelling of Software Architectures: Importance, Notions, Experiences

1990-22 H. Fassbender, H. Vogler: A Call-by-need Implementation of Syntax Directed Functional Programming

1991-01 Guenther Geiler (ed.), Fachgruppe Informatik: Jahresbericht 1990

1991-03 B. Steffen, A. Ingolfsdottir: Characteristic Formulae for Processes with Divergence

1991-04 M. Portz: A new class of cryptosystems based on interconnection networks

1991-05 H. Kuchen, G. Geiler: Distributed Applicative Arrays
1991-06 * Ludwig Staiger: Kolmogorov Complexity and Hausdorff Dimension
1991-07 * Ludwig Staiger: Syntactic Congruences for w-languages
1991-09 * Eila Kuikka: A Proposal for a Syntax-Directed Text Processing System
1991-10 K. Gladitz, H. Fassbender, H. Vogler: Compiler-based Implementation of Syntax-Directed Functional Programming
1991-11 R. Loogen, St. Winkler: Dynamic Detection of Determinism in Functional Logic Languages
1991-12 * K. Indermark, M. Rodriguez Artalejo (Eds.): Granada Workshop on the Integration of Functional and Logic Programming
1991-13 * Rolf Hager, Wolfgang Kremer: The Adaptive Priority Scheduler: A More Fair Priority Service Discipline
1991-14 * Andreas Fasbender, Wolfgang Kremer: A New Approximation Algorithm for Tandem Networks with Priority Nodes
1991-15 J. Börstler, A. Zündorf: Revisiting extensions to Modula-2 to support reusability
1991-16 J. Börstler, Th. Janning: Bridging the gap between Requirements Analysis and Design
1991-17 A. Zündorf, A. Schürr: Nondeterministic Control Structures for Graph Rewriting Systems
1991-18 * Matthias Jarke, John Mylopoulos, Joachim W. Schmidt, Yannis Vassiliou: DAIDA: An Environment for Evolving Information Systems
1991-19 M. Jeusfeld, M. Jarke: From Relational to Object-Oriented Integrity Simplification
1991-20 G. Hogen, A. Kindler, R. Loogen: Automatic Parallelization of Lazy Functional Programs
1991-21 * Prof. Dr. rer. nat. Otto Spaniol: ODP (Open Distributed Processing): Yet another Viewpoint
1991-22 H. Kuchen, F. Lücking, H. Stoltze: The Topology Description Language TDL
1991-23 S. Graf, B. Steffen: Compositional Minimization of Finite State Systems
1991-24 R. Cleaveland, J. Parrow, B. Steffen: The Concurrency Workbench: A Semantics Based Tool for the Verification of Concurrent Systems
1991-25 * Rudolf Mathar, Jürgen Mattfeldt: Optimal Transmission Ranges for Mobile Communication in Linear Multihop Packet Radio Networks
1991-26 M. Jeusfeld, M. Staudt: Query Optimization in Deductive Object Bases
1991-27 J. Knoop, B. Steffen: The Interprocedural Coincidence Theorem
1991-28 J. Knoop, B. Steffen: Unifying Strength Reduction and Semantic Code Motion
1991-30 T. Margaria: First-Order theories for the verification of complex FSMs
1991-31 B. Steffen: Generating Data Flow Analysis Algorithms from Modal Specifications
1992-01 Stefan Eherer (ed.), Fachgruppe Informatik: Jahresbericht 1991
1992-02 * Bernhard Westfechtel: Basismechanismen zur Datenverwaltung in strukturbezogenen Hypertextsystemen
1992-04 S. A. Smolka, B. Steffen: Priority as Extremal Probability
1992-05 * Matthias Jarke, Carlos Maltzahn, Thomas Rose: Sharing Processes: Team Coordination in Design Repositories

1992-06 O. Burkart, B. Steffen: Model Checking for Context-Free Processes

1992-07 * Matthias Jarke, Klaus Pohl: Information Systems Quality and Quality Information Systems

1992-08 * Rudolf Mathar, Jürgen Mattfeldt: Analyzing Routing Strategy NFP in Multihop Packet Radio Networks on a Line

1992-09 * Alfons Kemper, Guido Moerkotte: Grundlagen objektorientierter Datenbanksysteme

1992-10 Matthias Jarke, Manfred Jeusfeld, Andreas Miethsam, Michael Gocek: Towards a logic-based reconstruction of software configuration management

1992-11 Werner Hans: A Complete Indexing Scheme for WAM-based Abstract Machines

1992-12 W. Hans, R. Loogen, St. Winkler: On the Interaction of Lazy Evaluation and Backtracking

1992-13 * Matthias Jarke, Thomas Rose: Specification Management with CAD

1992-14 Th. Noll, H. Vogler: Top-down Parsing with Simultaneous Evaluation on Noncircular Attribute Grammars

1992-15 A. Schuerr, B. Westfechtel: Graphgrammatiken und Graphersetzungssysteme(written in german)

1992-16 * Graduiertenkolleg Informatik und Technik (Hrsg.): Forschungsprojekte des Graduiertenkollegs Informatik und Technik

1992-17 M. Jarke (ed.): ConceptBase V3.1 User Manual

1992-18 * Clarence A. Ellis, Matthias Jarke (Eds.): Distributed Cooperation in Integrated Information Systems - Proceedings of the Third International Workshop on Intelligent and Cooperative Information Systems

1992-19-00 H. Kuchen, R. Loogen (eds.): Proceedings of the 4th Int. Workshop on the Parallel Implementation of Functional Languages

1992-19-01 G. Hogen, R. Loogen: PASTEL - A Parallel Stack-Based Implementation of Eager Functional Programs with Lazy Data Structures (Extended Abstract)

1992-19-02 H. Kuchen, K. Gladitz: Implementing Bags on a Shared Memory MIMD-Machine

1992-19-03 C. Rathsack, S.B. Scholz: LISA - A Lazy Interpreter for a Full-Fledged Lambda-Calculus

1992-19-04 T.A. Bratvold: Determining Useful Parallelism in Higher Order Functions

1992-19-05 S. Kahrs: Polymorphic Type Checking by Interpretation of Code

1992-19-06 M. Chakravarty, M. Köhler: Equational Constraints, Residuation, and the Parallel JUMP-Machine

1992-19-07 J. Seward: Polymorphic Strictness Analysis using Frontiers (Draft Version)

1992-19-08 D. Gärtner, A. Kimms, W. Kluge: pi-Red^+ - A Compiling Graph-Reduction System for a Full Fledged Lambda-Calculus

1992-19-09 D. Howe, G. Burn: Experiments with strict STG code

1992-19-10 J. Glauert: Parallel Implementation of Functional Languages Using Small Processes

1992-19-11 M. Joy, T. Axford: A Parallel Graph Reduction Machine

1992-19-12 A. Bennett, P. Kelly: Simulation of Multicache Parallel Reduction

1992-19-13 K. Langendoen, D.J. Agterkamp: Cache Behaviour of Lazy Functional Programs (Working Paper)

1992-19-14 K. Hammond, S. Peyton Jones: Profiling scheduling strategies on the GRIP parallel reducer

1992-19-15 S. Mintchev: Using Strictness Information in the STG-machine

1992-19-16 D. Rushall: An Attribute Grammar Evaluator in Haskell

1992-19-17 J. Wild, H. Glaser, P. Hartel: Statistics on storage management in a lazy functional language implementation

1992-19-18 W.S. Martins: Parallel Implementations of Functional Languages

1992-19-19 D. Lester: Distributed Garbage Collection of Cyclic Structures (Draft version)

1992-19-20 J.C. Glas, R.F.H. Hofman, W.G. Vree: Parallelization of Branch-and-Bound Algorithms in a Functional Programming Environment

1992-19-21 S. Hwang, D. Rushall: The nu-STG machine: a parallelized Spineless Tagless Graph Reduction Machine in a distributed memory architecture (Draft version)

1992-19-22 G. Burn, D. Le Metayer: Cps-Translation and the Correctness of Optimising Compilers

1992-19-23 S.L. Peyton Jones, P. Wadler: Imperative functional programming (Brief summary)

1992-19-24 W. Damm, F. Liu, Th. Peikenkamp: Evaluation and Parallelization of Functions in Functional + Logic Languages (abstract)

1992-19-25 M. Kesseler: Communication Issues Regarding Parallel Functional Graph Rewriting

1992-19-26 Th. Peikenkamp: Charakterizing and representing neededness in functional loginc languages (abstract)

1992-19-27 H. Doerr: Monitoring with Graph-Grammars as formal operational Models

1992-19-28 J. van Groningen: Some implementation aspects of Concurrent Clean on distributed memory architectures

1992-19-29 G. Ostheimer: Load Bounding for Implicit Parallelism (abstract)

1992-20 H. Kuchen, F.J. Lopez Fraguas, J.J. Moreno Navarro, M. Rodriguez Artalejo: Implementing Disequality in a Lazy Functional Logic Language

1992-21 H. Kuchen, F.J. Lopez Fraguas: Result Directed Computing in a Functional Logic Language

1992-22 H. Kuchen, J.J. Moreno Navarro, M.V. Hermenegildo: Independent AND-Parallel Narrowing

1992-23 T. Margaria, B. Steffen: Distinguishing Formulas for Free

1992-24 K. Pohl: The Three Dimensions of Requirements Engineering

1992-25 * R. Stainov: A Dynamic Configuration Facility for Multimedia Communications

1992-26 * Michael von der Beeck: Integration of Structured Analysis and Timed Statecharts for Real-Time and Concurrency Specification

1992-27 W. Hans, St. Winkler: Aliasing and Groundness Analysis of Logic Programs through Abstract Interpretation and its Safety

1992-28 * Gerhard Steinke, Matthias Jarke: Support for Security Modeling in Information Systems Design

1992-29 B. Schinzel: Warum Frauenforschung in Naturwissenschaft und Technik

1992-30 A. Kemper, G. Moerkotte, K. Peithner: Object-Orientation Axiomatised by Dynamic Logic

1992-32 * Bernd Heinrichs, Kai Jakobs: Timer Handling in High-Performance Transport Systems

1992-33 * B. Heinrichs, K. Jakobs, K. Lenßen, W. Reinhardt, A. Spinner: Euro-Bridge: Communication Services for Multimedia Applications

1992-34 C. Gerlhof, A. Kemper, Ch. Kilger, G. Moerkotte: Partition-Based Clustering in Object Bases: From Theory to Practice

1992-35 J. Börstler: Feature-Oriented Classification and Reuse in IPSEN

1992-36 M. Jarke, J. Bubenko, C. Rolland, A. Sutcliffe, Y. Vassiliou: Theories Underlying Requirements Engineering: An Overview of NATURE at Genesis

1992-37 * K. Pohl, M. Jarke: Quality Information Systems: Repository Support for Evolving Process Models

1992-38 A. Zuendorf: Implementation of the imperative / rule based language PROGRES

1992-39 P. Koch: Intelligentes Backtracking bei der Auswertung funktional-logischer Programme

1992-40 * Rudolf Mathar, Jürgen Mattfeldt: Channel Assignment in Cellular Radio Networks

1992-41 * Gerhard Friedrich, Wolfgang Neidl: Constructive Utility in Model-Based Diagnosis Repair Systems

1992-42 * P. S. Chen, R. Hennicker, M. Jarke: On the Retrieval of Reusable Software Components

1992-43 W. Hans, St.Winkler: Abstract Interpretation of Functional Logic Languages

1992-44 N. Kiesel, A. Schuerr, B. Westfechtel: Design and Evaluation of GRAS, a Graph-Oriented Database System for Engineering Applications

1993-01 * Fachgruppe Informatik: Jahresbericht 1992

1993-02 * Patrick Shicheng Chen: On Inference Rules of Logic-Based Information Retrieval Systems

1993-03 G. Hogen, R. Loogen: A New Stack Technique for the Management of Runtime Structures in Distributed Environments

1993-05 A. Zündorf: A Heuristic for the Subgraph Isomorphism Problem in Executing PROGRES

1993-06 A. Kemper, D. Kossmann: Adaptable Pointer Swizzling Strategies in Object Bases: Design, Realization, and Quantitative Analysis

1993-07 * Graduiertenkolleg Informatik und Technik (Hrsg.): Graduiertenkolleg Informatik und Technik

1993-08 * Matthias Berger: k-Coloring Vertices using a Neural Network with Convergence to Valid Solutions

1993-09 M. Buchheit, M. Jeusfeld, W. Nutt, M. Staudt: Subsumption between Queries to Object-Oriented Databases

1993-10 O. Burkart, B. Steffen: Pushdown Processes: Parallel Composition and Model Checking

1993-11 * R. Große-Wienker, O. Hermanns, D. Menzenbach, A. Pollacks, S. Repetzki, J. Schwartz, K. Sonnenschein, B. Westfechtel: Das SUKITS-Projekt: A-posteriori-Integration heterogener CIM-Anwendungssysteme

1993-12 * Rudolf Mathar, Jürgen Mattfeldt: On the Distribution of Cumulated Interference Power in Rayleigh Fading Channels
1993-13 O. Maler, L. Staiger: On Syntactic Congruences for omega-languages
1993-14 M. Jarke, St. Eherer, R. Gallersdoerfer, M. Jeusfeld, M. Staudt: ConceptBase - A Deductive Object Base Manager
1993-15 M. Staudt, H.W. Nissen, M.A. Jeusfeld: Query by Class, Rule and Concept
1993-16 * M. Jarke, K. Pohl, St. Jacobs et al.: Requirements Engineering: An Integrated View of Representation Process and Domain
1993-17 * M. Jarke, K. Pohl: Establishing Vision in Context: Towards a Model of Requirements Processes
1993-18 W. Hans, H. Kuchen, St. Winkler: Full Indexing for Lazy Narrowing
1993-19 W. Hans, J.J. Ruz, F. Saenz, St. Winkler: A VHDL Specification of a Shared Memory Parallel Machine for Babel
1993-20 * K. Finke, M. Jarke, P. Szczurko, R. Soltysiak: Quality Management for Expert Systems in Process Control
1993-21 M. Jarke, M.A. Jeusfeld, P. Szczurko: Three Aspects of Intelligent Cooperation in the Quality Cycle
1994-01 Margit Generet, Sven Martin (eds.), Fachgruppe Informatik: Jahresbericht 1993
1994-02 M. Lefering: Development of Incremental Integration Tools Using Formal Specifications
1994-03 * P. Constantopoulos, M. Jarke, J. Mylopoulos, Y. Vassiliou: The Software Information Base: A Server for Reuse
1994-04 * Rolf Hager, Rudolf Mathar, Jürgen Mattfeldt: Intelligent Cruise Control and Reliable Communication of Mobile Stations
1994-05 * Rolf Hager, Peter Hermesmann, Michael Portz: Feasibility of Authentication Procedures within Advanced Transport Telematics
1994-06 * Claudia Popien, Bernd Meyer, Axel Kuepper: A Formal Approach to Service Import in ODP Trader Federations
1994-07 P. Peters, P. Szczurko: Integrating Models of Quality Management Methods by an Object-Oriented Repository
1994-08 * Manfred Nagl, Bernhard Westfechtel: A Universal Component for the Administration in Distributed and Integrated Development Environments
1994-09 * Patrick Horster, Holger Petersen: Signatur- und Authentifikationsverfahren auf der Basis des diskreten Logarithmusproblems
1994-11 A. Schürr: PROGRES, A Visual Language and Environment for PROgramming with Graph REwrite Systems
1994-12 A. Schürr: Specification of Graph Translators with Triple Graph Grammars
1994-13 A. Schürr: Logic Based Programmed Structure Rewriting Systems
1994-14 L. Staiger: Codes, Simplifying Words, and Open Set Condition
1994-15 * Bernhard Westfechtel: A Graph-Based System for Managing Configurations of Engineering Design Documents
1994-16 P. Klein: Designing Software with Modula-3
1994-17 I. Litovsky, L. Staiger: Finite acceptance of infinite words

1994-18 G. Hogen, R. Loogen: Parallel Functional Implementations: Graphbased vs. Stackbased Reduction

1994-19 M. Jeusfeld, U. Johnen: An Executable Meta Model for Re-Engineering of Database Schemas

1994-20 * R. Gallersdörfer, M. Jarke, K. Klabunde: Intelligent Networks as a Data Intensive Application (INDIA)

1994-21 M. Mohnen: Proving the Correctness of the Static Link Technique Using Evolving Algebras

1994-22 H. Fernau, L. Staiger: Valuations and Unambiguity of Languages, with Applications to Fractal Geometry

1994-24 * M. Jarke, K. Pohl, R. Dömges, St. Jacobs, H. W. Nissen: Requirements Information Management: The NATURE Approach

1994-25 * M. Jarke, K. Pohl, C. Rolland, J.-R. Schmitt: Experience-Based Method Evaluation and Improvement: A Process Modeling Approach

1994-26 * St. Jacobs, St. Kethers: Improving Communication and Decision Making within Quality Function Deployment

1994-27 * M. Jarke, H. W. Nissen, K. Pohl: Tool Integration in Evolving Information Systems Environments

1994-28 O. Burkart, D. Caucal, B. Steffen: An Elementary Bisimulation Decision Procedure for Arbitrary Context-Free Processes

1995-01 * Fachgruppe Informatik: Jahresbericht 1994

1995-02 Andy Schürr, Andreas J. Winter, Albert Zündorf: Graph Grammar Engineering with PROGRES

1995-03 Ludwig Staiger: A Tight Upper Bound on Kolmogorov Complexity by Hausdorff Dimension and Uniformly Optimal Prediction

1995-04 Birgitta König-Ries, Sven Helmer, Guido Moerkotte: An experimental study on the complexity of left-deep join ordering problems for cyclic queries

1995-05 Sophie Cluet, Guido Moerkotte: Efficient Evaluation of Aggregates on Bulk Types

1995-06 Sophie Cluet, Guido Moerkotte: Nested Queries in Object Bases

1995-07 Sophie Cluet, Guido Moerkotte: Query Optimization Techniques Exploiting Class Hierarchies

1995-08 Markus Mohnen: Efficient Compile-Time Garbage Collection for Arbitrary Data Structures

1995-09 Markus Mohnen: Functional Specification of Imperative Programs: An Alternative Point of View of Functional Languages

1995-10 Rainer Gallersdörfer, Matthias Nicola: Improving Performance in Replicated Databases through Relaxed Coherency

1995-11 * M.Staudt, K.von Thadden: Subsumption Checking in Knowledge Bases

1995-12 * G.V.Zemanek, H.W.Nissen, H.Hubert, M.Jarke: Requirements Analysis from Multiple Perspectives: Experiences with Conceptual Modeling Technology

1995-13 * M.Staudt, M.Jarke: Incremental Maintenance of Externally Materialized Views

1995-14 * P.Peters, P.Szczurko, M.Jeusfeld: Oriented Information Management: Conceptual Models at Work

1995-15 * Matthias Jarke, Sudha Ram (Hrsg.): WITS 95 Proceedings of the 5th Annual Workshop on Information Technologies and Systems

1995-16 * W.Hans, St.Winkler, F.Saenz: Distributed Execution in Functional Logic Programming

1996-01 * Jahresbericht 1995

1996-02 Michael Hanus, Christian Prehofer: Higher-Order Narrowing with Definitional Trees

1996-03 * W.Scheufele, G.Moerkotte: Optimal Ordering of Selections and Joins in Acyclic Queries with Expensive Predicates

1996-04 Klaus Pohl: PRO-ART: Enabling Requirements Pre-Traceability

1996-05 Klaus Pohl: Requirements Engineering: An Overview

1996-06 * M.Jarke, W.Marquardt: Design and Evaluation of Computer–Aided Process Modelling Tools

1996-07 Olaf Chitil: The Sigma-Semantics: A Comprehensive Semantics for Functional Programs

1996-08 * S.Sripada: On Entropy and the Limitations of the Second Law of Thermodynamics

1996-09 Michael Hanus (Ed.): Proceedings of the Poster Session of ALP96 - Fifth International Conference on Algebraic and Logic Programming

1996-09-0 Michael Hanus (Ed.): Proceedings of the Poster Session of ALP 96 - Fifth International Conference on Algebraic and Logic Programming: Introduction and table of contents

1996-09-1 Ilies Alouini: An Implementation of Conditional Concurrent Rewriting on Distributed Memory Machines

1996-09-2 Olivier Danvy, Karoline Malmkjær: On the Idempotence of the CPS Transformation

1996-09-3 Víctor M. Gulías, José L. Freire: Concurrent Programming in Haskell

1996-09-4 Sébastien Limet, Pierre Réty: On Decidability of Unifiability Modulo Rewrite Systems

1996-09-5 Alexandre Tessier: Declarative Debugging in Constraint Logic Programming

1996-10 Reidar Conradi, Bernhard Westfechtel: Version Models for Software Configuration Management

1996-11 * C.Weise, D.Lenzkes: A Fast Decision Algorithm for Timed Refinement

1996-12 * R.Dömges, K.Pohl, M.Jarke, B.Lohmann, W.Marquardt: PRO-ART/CE* — An Environment for Managing the Evolution of Chemical Process Simulation Models

1996-13 * K.Pohl, R.Klamma, K.Weidenhaupt, R.Dömges, P.Haumer, M.Jarke: A Framework for Process-Integrated Tools

1996-14 * R.Gallersdörfer, K.Klabunde, A.Stolz, M.Eßmajor: INDIA — Intelligent Networks as a Data Intensive Application, Final Project Report, June 1996

1996-15 * H.Schimpe, M.Staudt: VAREX: An Environment for Validating and Refining Rule Bases

1996-16 * M.Jarke, M.Gebhardt, S.Jacobs, H.Nissen: Conflict Analysis Across Heterogeneous Viewpoints: Formalization and Visualization

1996-17 Manfred A. Jeusfeld, Tung X. Bui: Decision Support Components on the Internet

1996-18 Manfred A. Jeusfeld, Mike Papazoglou: Information Brokering: Design, Search and Transformation

1996-19 * P.Peters, M.Jarke: Simulating the impact of information flows in networked organizations

1996-20 Matthias Jarke, Peter Peters, Manfred A. Jeusfeld: Model-driven planning and design of cooperative information systems

1996-21 * G.de Michelis, E.Dubois, M.Jarke, F.Matthes, J.Mylopoulos, K.Pohl, J.Schmidt, C.Woo, E.Yu: Cooperative information systems: a manifesto

1996-22 * S.Jacobs, M.Gebhardt, S.Kethers, W.Rzasa: Filling HTML forms simultaneously: CoWeb architecture and functionality

1996-23 * M.Gebhardt, S.Jacobs: Conflict Management in Design

1997-01 Michael Hanus, Frank Zartmann (eds.): Jahresbericht 1996

1997-02 Johannes Faassen: Using full parallel Boltzmann Machines for Optimization

1997-03 Andreas Winter, Andy Schürr: Modules and Updatable Graph Views for PROgrammed Graph REwriting Systems

1997-04 Markus Mohnen, Stefan Tobies: Implementing Context Patterns in the Glasgow Haskell Compiler

1997-05 * S.Gruner: Schemakorrespondenzaxiome unterstützen die paargrammatische Spezifikation inkrementeller Integrationswerkzeuge

1997-06 Matthias Nicola, Matthias Jarke: Design and Evaluation of Wireless Health Care Information Systems in Developing Countries

1997-07 Petra Hofstedt: Taskparallele Skelette für irregulär strukturierte Probleme in deklarativen Sprachen

1997-08 Dorothea Blostein, Andy Schürr: Computing with Graphs and Graph Rewriting

1997-09 Carl-Arndt Krapp, Bernhard Westfechtel: Feedback Handling in Dynamic Task Nets

1997-10 Matthias Nicola, Matthias Jarke: Integrating Replication and Communication in Performance Models of Distributed Databases

1997-11 * R. Klamma, P. Peters, M. Jarke: Workflow Support for Failure Management in Federated Organizations

1997-13 Markus Mohnen: Optimising the Memory Management of Higher-Order Functional Programs

1997-14 Roland Baumann: Client/Server Distribution in a Structure-Oriented Database Management System

1997-15 George Botorog: High-Level Parallel Programming and the Efficient Implementation of Numerical Algorithms

1998-01 * Fachgruppe Informatik: Jahresbericht 1997

1998-02 Stefan Gruner, Manfred Nagel, Andy Schürr: Fine-grained and Structure-Oriented Document Integration Tools are Needed for Development Processes

1998-03 Stefan Gruner: Einige Anmerkungen zur graphgrammatischen Spezifikation von Integrationswerkzeugen nach Westfechtel, Janning, Lefering und Schürr

1998-04 * O. Kubitz: Mobile Robots in Dynamic Environments

1998-05 Martin Leucker, Stephan Tobies: Truth - A Verification Platform for Distributed Systems

1998-06 * Matthias Oliver Berger: DECT in the Factory of the Future

1998-07 M. Arnold, M. Erdmann, M. Glinz, P. Haumer, R. Knoll, B. Paech, K. Pohl, J. Ryser, R. Studer, K. Weidenhaupt: Survey on the Scenario Use in Twelve Selected Industrial Projects

1998-09 * Th. Lehmann: Geometrische Ausrichtung medizinischer Bilder am Beispiel intraoraler Radiographien

1998-10 * M. Nicola, M. Jarke: Performance Modeling of Distributed and Replicated Databases

1998-11 * Ansgar Schleicher, Bernhard Westfechtel, Dirk Jäger: Modeling Dynamic Software Processes in UML

1998-12 * W. Appelt, M. Jarke: Interoperable Tools for Cooperation Support using the World Wide Web

1998-13 Klaus Indermark: Semantik rekursiver Funktionsdefinitionen mit Striktheitsinformation

1999-01 * Jahresbericht 1998

1999-02 * F. Huch: Verifcation of Erlang Programs using Abstract Interpretation and Model Checking — Extended Version

1999-03 * R. Gallersdörfer, M. Jarke, M. Nicola: The ADR Replication Manager

1999-04 María Alpuente, Michael Hanus, Salvador Lucas, Germán Vidal: Specialization of Functional Logic Programs Based on Needed Narrowing

1999-05 * W. Thomas (Ed.): DLT 99 - Developments in Language Theory Fourth International Conference

1999-06 * Kai Jakobs, Klaus-Dieter Kleefeld: Informationssysteme für die angewandte historische Geographie

1999-07 Thomas Wilke: CTL+ is exponentially more succinct than CTL

1999-08 Oliver Matz: Dot-Depth and Monadic Quantifier Alternation over Pictures

2000-01 * Jahresbericht 1999

2000-02 Jens Vöge, Marcin Jurdzinski A Discrete Strategy Improvement Algorithm for Solving Parity Games

2000-03 D. Jäger, A. Schleicher, B. Westfechtel: UPGRADE: A Framework for Building Graph-Based Software Engineering Tools

2000-04 Andreas Becks, Stefan Sklorz, Matthias Jarke: Exploring the Semantic Structure of Technical Document Collections: A Cooperative Systems Approach

2000-05 Mareike Schoop: Cooperative Document Management

2000-06 Mareike Schoop, Christoph Quix (eds.): Proceedings of the Fifth International Workshop on the Language-Action Perspective on Communication Modelling

2000-07 * Markus Mohnen, Pieter Koopman (Eds.): Proceedings of the 12th International Workshop of Functional Languages

2000-08 Thomas Arts, Thomas Noll: Verifying Generic Erlang Client-Server Implementations

2001-01 * Jahresbericht 2000

2001-02 Benedikt Bollig, Martin Leucker: Deciding LTL over Mazurkiewicz Traces

2001-03 Thierry Cachat: The power of one-letter rational languages

2001-04 Benedikt Bollig, Martin Leucker, Michael Weber: Local Parallel Model Checking for the Alternation Free mu-Calculus
2001-05 Benedikt Bollig, Martin Leucker, Thomas Noll: Regular MSC Languages
2001-06 Achim Blumensath: Prefix-Recognisable Graphs and Monadic Second-Order Logic
2001-07 Martin Grohe, Stefan Wöhrle: An Existential Locality Theorem
2001-08 Mareike Schoop, James Taylor (eds.): Proceedings of the Sixth International Workshop on the Language-Action Perspective on Communication Modelling
2001-09 Thomas Arts, Jürgen Giesl: A collection of examples for termination of term rewriting using dependency pairs
2001-10 Achim Blumensath: Axiomatising Tree-interpretable Structures
2001-11 Klaus Indermark, Thomas Noll (eds.): Kolloquium Programmiersprachen und Grundlagen der Programmierung
2002-01 * Jahresbericht 2001
2002-02 Jürgen Giesl, Aart Middeldorp: Transformation Techniques for Context-Sensitive Rewrite Systems
2002-03 Benedikt Bollig, Martin Leucker, Thomas Noll: Generalised Regular MSC Languages
2002-04 Jürgen Giesl, Aart Middeldorp: Innermost Termination of Context-Sensitive Rewriting
2002-05 Horst Lichter, Thomas von der Maßen, Thomas Weiler: Modelling Requirements and Architectures for Software Product Lines
2002-06 Henry N. Adorna: 3-Party Message Complexity is Better than 2-Party Ones for Proving Lower Bounds on the Size of Minimal Nondeterministic Finite Automata
2002-07 Jörg Dahmen: Invariant Image Object Recognition using Gaussian Mixture Densities
2002-08 Markus Mohnen: An Open Framework for Data-Flow Analysis in Java
2002-09 Markus Mohnen: Interfaces with Default Implementations in Java
2002-10 Martin Leucker: Logics for Mazurkiewicz traces
2002-11 Jürgen Giesl, Hans Zantema: Liveness in Rewriting
2003-01 * Jahresbericht 2002
2003-02 Jürgen Giesl, René Thiemann: Size-Change Termination for Term Rewriting
2003-03 Jürgen Giesl, Deepak Kapur: Deciding Inductive Validity of Equations
2003-04 Jürgen Giesl, René Thiemann, Peter Schneider-Kamp, Stephan Falke: Improving Dependency Pairs
2003-05 Christof Löding, Philipp Rohde: Solving the Sabotage Game is PSPACE-hard
2003-06 Franz Josef Och: Statistical Machine Translation: From Single-Word Models to Alignment Templates
2003-07 Horst Lichter, Thomas von der Maßen, Alexander Nyßen, Thomas Weiler: Vergleich von Ansätzen zur Feature Modellierung bei der Softwareproduktlinienentwicklung
2003-08 Jürgen Giesl, René Thiemann, Peter Schneider-Kamp, Stephan Falke: Mechanizing Dependency Pairs
2004-01 * Fachgruppe Informatik: Jahresbericht 2003

2004-02 Benedikt Bollig, Martin Leucker: Message-Passing Automata are expressively equivalent to EMSO logic
2004-03 Delia Kesner, Femke van Raamsdonk, Joe Wells (eds.): HOR 2004 – 2nd International Workshop on Higher-Order Rewriting
2004-04 Slim Abdennadher, Christophe Ringeissen (eds.): RULE 04 – Fifth International Workshop on Rule-Based Programming
2004-05 Herbert Kuchen (ed.): WFLP 04 – 13th International Workshop on Functional and (Constraint) Logic Programming
2004-06 Sergio Antoy, Yoshihito Toyama (eds.): WRS 04 – 4th International Workshop on Reduction Strategies in Rewriting and Programming
2004-07 Michael Codish, Aart Middeldorp (eds.): WST 04 – 7th International Workshop on Termination
2004-08 Klaus Indermark, Thomas Noll: Algebraic Correctness Proofs for Compiling Recursive Function Definitions with Strictness Information
2004-09 Joachim Kneis, Daniel Mölle, Stefan Richter, Peter Rossmanith: Parameterized Power Domination Complexity
2004-10 Zinaida Benenson, Felix C. Gärtner, Dogan Kesdogan: Secure Multi-Party Computation with Security Modules
2005-01 * Fachgruppe Informatik: Jahresbericht 2004
2005-02 Maximillian Dornseif, Felix C. Gärtner, Thorsten Holz, Martin Mink: An Offensive Approach to Teaching Information Security: "Aachen Summer School Applied IT Security"
2005-03 Jürgen Giesl, René Thiemann, Peter Schneider-Kamp: Proving and Disproving Termination of Higher-Order Functions
2005-04 Daniel Mölle, Stefan Richter, Peter Rossmanith: A Faster Algorithm for the Steiner Tree Problem
2005-05 Fabien Pouget, Thorsten Holz: A Pointillist Approach for Comparing Honeypots
2005-06 Simon Fischer, Berthold Vöcking: Adaptive Routing with Stale Information
2005-07 Felix C. Freiling, Thorsten Holz, Georg Wicherski: Botnet Tracking: Exploring a Root-Cause Methodology to Prevent Distributed Denial-of-Service Attacks
2005-08 Joachim Kneis, Peter Rossmanith: A New Satisfiability Algorithm With Applications To Max-Cut
2005-09 Klaus Kursawe, Felix C. Freiling: Byzantine Fault Tolerance on General Hybrid Adversary Structures
2005-10 Benedikt Bollig: Automata and Logics for Message Sequence Charts
2005-11 Simon Fischer, Berthold Vöcking: A Counterexample to the Fully Mixed Nash Equilibrium Conjecture
2005-12 Neeraj Mittal, Felix Freiling, S. Venkatesan, Lucia Draque Penso: Efficient Reductions for Wait-Free Termination Detection in Faulty Distributed Systems
2005-13 Carole Delporte-Gallet, Hugues Fauconnier, Felix C. Freiling: Revisiting Failure Detection and Consensus in Omission Failure Environments
2005-14 Felix C. Freiling, Sukumar Ghosh: Code Stabilization
2005-15 Uwe Naumann: The Complexity of Derivative Computation

2005-16 Uwe Naumann: Syntax-Directed Derivative Code (Part I: Tangent-Linear Code)

2005-17 Uwe Naumann: Syntax-directed Derivative Code (Part II: Intraprocedural Adjoint Code)

2005-18 Thomas von der Maßen, Klaus Müller, John MacGregor, Eva Geisberger, Jörg Dörr, Frank Houdek, Harbhajan Singh, Holger Wußmann, Hans-Veit Bacher, Barbara Paech: Einsatz von Features im Software-Entwicklungsprozess - Abschlußbericht des GI-Arbeitskreises "Features"

2005-19 Uwe Naumann, Andre Vehreschild: Tangent-Linear Code by Augmented LL-Parsers

2005-20 Felix C. Freiling, Martin Mink: Bericht über den Workshop zur Ausbildung im Bereich IT-Sicherheit Hochschulausbildung, berufliche Weiterbildung, Zertifizierung von Ausbildungsangeboten am 11. und 12. August 2005 in Köln organisiert von RWTH Aachen in Kooperation mit BITKOM, BSI, DLR und Gesellschaft fuer Informatik (GI) e.V.

2005-21 Thomas Noll, Stefan Rieger: Optimization of Straight-Line Code Revisited

2005-22 Felix Freiling, Maurice Herlihy, Lucia Draque Penso: Optimal Randomized Fair Exchange with Secret Shared Coins

2005-23 Heiner Ackermann, Alantha Newman, Heiko Röglin, Berthold Vöcking: Decision Making Based on Approximate and Smoothed Pareto Curves

2005-24 Alexander Becher, Zinaida Benenson, Maximillian Dornseif: Tampering with Motes: Real-World Physical Attacks on Wireless Sensor Networks

2006-01 * Fachgruppe Informatik: Jahresbericht 2005

2006-03 Michael Maier, Uwe Naumann: Intraprocedural Adjoint Code Generated by the Differentiation-Enabled NAGWare Fortran Compiler

2006-04 Ebadollah Varnik, Uwe Naumann, Andrew Lyons: Toward Low Static Memory Jacobian Accumulation

2006-05 Uwe Naumann, Jean Utke, Patrick Heimbach, Chris Hill, Derya Ozyurt, Carl Wunsch, Mike Fagan, Nathan Tallent, Michelle Strout: Adjoint Code by Source Transformation with OpenAD/F

2006-06 Joachim Kneis, Daniel Mölle, Stefan Richter, Peter Rossmanith: Divide-and-Color

2006-07 Thomas Colcombet, Christof Löding: Transforming structures by set interpretations

2006-08 Uwe Naumann, Yuxiao Hu: Optimal Vertex Elimination in Single-Expression-Use Graphs

2006-09 Tingting Han, Joost-Pieter Katoen: Counterexamples in Probabilistic Model Checking

2006-10 Mesut Günes, Alexander Zimmermann, Martin Wenig, Jan Ritzerfeld, Ulrich Meis: From Simulations to Testbeds - Architecture of the Hybrid MCG-Mesh Testbed

2006-11 Bastian Schlich, Michael Rohrbach, Michael Weber, Stefan Kowalewski: Model Checking Software for Microcontrollers

2006-12 Benedikt Bollig, Joost-Pieter Katoen, Carsten Kern, Martin Leucker: Replaying Play in and Play out: Synthesis of Design Models from Scenarios by Learning

2006-13 Wong Karianto, Christof Löding: Unranked Tree Automata with Sibling Equalities and Disequalities

2006-14 Danilo Beuche, Andreas Birk, Heinrich Dreier, Andreas Fleischmann, Heidi Galle, Gerald Heller, Dirk Janzen, Isabel John, Ramin Tavakoli Kolagari, Thomas von der Maßen, Andreas Wolfram: Report of the GI Work Group "Requirements Management Tools for Product Line Engineering"

2006-15 Sebastian Ullrich, Jakob T. Valvoda, Torsten Kuhlen: Utilizing optical sensors from mice for new input devices

2006-16 Rafael Ballagas, Jan Borchers: Selexels: a Conceptual Framework for Pointing Devices with Low Expressiveness

* These reports are only available as a printed version.
Please contact biblio@informatik.rwth-aachen.de to obtain copies.