

Dynamic Mapping of Physical Controls for Tabletop Groupware

Rebecca Fiebrink
Princeton University
Princeton, NJ, USA
fiebrink@princeton.edu

Dan Morris
Microsoft Research
Redmond, WA, USA
dan@microsoft.com

Meredith Ringel Morris
Microsoft Research
Redmond, WA, USA
merrie@microsoft.com

ABSTRACT

Multi-touch interactions are a promising means of control for interactive tabletops. However, a lack of precision and tactile feedback makes multi-touch controls a poor fit for tasks where precision and feedback are crucial. We present an approach that offers precise control and tactile feedback for tabletop systems through the integration of dynamically re-mappable physical controllers with the multi-touch environment, and we demonstrate this approach in our collaborative tabletop audio editing environment. An observational user study demonstrates that our approach can provide needed precision and feedback, while preserving the collaborative benefits of a shared direct-manipulation surface. Our observations also suggest that direct touch and physical controllers can offer complementary benefits, and that providing both allows users to adjust their control strategy based on considerations including precision, convenience, visibility, and user role.

Author Keywords

Interactive tabletops, surface computing.

ACM Classification Keywords

H5.3. Information interfaces and presentation (e.g., HCI): Group and Organization Interfaces – *computer-supported cooperative work*.

INTRODUCTION

Single display groupware (SDG) systems [26] allow multiple co-located users to collaborate by sharing control over a single display. Tabletop systems controlled by multi-touch gestures represent an exciting step toward wide availability of SDG, and such systems are now being commercially manufactured (e.g., [5,15]). Multi-touch control allows simultaneous direct manipulation by multiple users, and the rich expressiveness of multi-touch gestures allows for natural and intuitive interactions [31]. However, the precision of multi-touch control is limited by factors including the sensing resolution and noise of the underlying technology [3], the size of the fingertip itself [23] (which is significantly larger than a stylus or mouse cursor), and the

fact that the sensed boundary of the fingertip may vary appreciably with finger angle [8]. The precision of touch-based systems is further limited by a lack of feedback: a touch-sensitive surface provides very limited haptic feedback, and visual feedback may be impaired by the finger occluding the object under manipulation [29]. Furthermore, the visibility of gesture—a core benefit of multi-touch control in a collaborative setting—may be diminished when the physical manipulations are small in magnitude. For these reasons, multi-touch interactions may be a poor choice when fine-grained control is necessary.

Motivated by the challenge of allowing greater precision while also supporting collaboration, we propose an approach to integrating dynamically mapped physical controllers with a shared multi-touch tabletop. Specifically, we allow users to dynamically bind application parameters to both *virtual* controls (UI widgets) and *physical* controls (knobs and buttons) that offer precision and feedback for fine-grained manipulations. Physical controls are deployed in the form of several small, repositionable boxes that can be distributed among users, providing individuals with their own replications of a subset of the available virtual controls (Figure 1). We provide users with mechanisms to dynamically map these physical controls to the available parameters, allowing a flexible tailoring of the mapping state to the group interaction style and task requirements. This approach is applicable to an array of tasks requiring fine-grained control over many parameters and requiring the collaboration of multiple users, such as media editing,



Figure 1: Dynamically re-mappable physical controllers are integrated into a multi-touch, collaborative tabletop environment.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2009, April 4–9, 2009, Boston, Massachusetts, USA.
Copyright 2009 ACM 978-1-60558-246-7/09/04...\$5.00

gaming, graphic design and layout, and data visualization. In order to investigate its efficacy, we have implemented this approach in the context of a collaborative audio editing environment.

The contributions of this work include techniques for integrating high-precision physical controllers into a collaborative tabletop environment, an observational user study demonstrating the effectiveness of these techniques and exploring the factors influencing users' preferences for physical or direct-touch controls and for local or central controls, and a discussion of our findings and their implications for direct-touch tabletop systems supporting precision-sensitive collaborative tasks.

RELATED WORK

Many strategies for increasing the precision of touch-controlled user interfaces have been proposed, including in-situ zooming [1,18], the exploitation of additional fingers [3], and the appearance of non-occluded callouts for small targets [29,31]. Such work has not explicitly considered the implications for co-located, collaborating users. On the other hand, there exists a growing body of work around interface design for co-located tabletop groupware, addressing fundamental questions regarding how input modality and interface layout can affect efficiency and the collaborative experience. Notably, Ha et al. [10] investigated the effects of direct (touch or stylus) and indirect (mouse) input on co-located users' experiences, and shed light on the tradeoffs inherent in each modality. For example, direct input allows for control gestures that are noticeable and understandable to users, but can also induce fear of physical collision or crossing into another user's territory, and present difficulty in reaching far-away items.

Interface layout merits particular consideration in the design of tabletop groupware. The DiamondSpin toolkit [25], for example, accommodates a single shared interface, multiple replicated interfaces, or both. Morris et al. [16] suggest that, in a touch-controlled multi-user tabletop system, users may prefer replicated controls (i.e., one copy of controls per user) over a shared central set of controls. This preference for replicated controls may be influenced by users' tendency to treat the portion of the table nearest them as their own territory [24]. However, simply replicating controls presents a challenge when the number of replicated controls is large, and can lead to controls becoming smaller (and therefore harder to manipulate precisely) in order to fit on the table.

Physical artifacts have been integrated into tabletop and other computing environments with a variety of goals. Physical artifacts may be employed as tangible props, in an effort to endow computer interfaces with the rich interactions afforded by physical objects [13]. For example, the Metadesk [28] uses physical representations of buildings to display, scale, and rotate a map, and the Actuated Workbench [19] allows a tabletop to provide physical outputs by automatically moving electromagnets

on its surface. Alternatively, more general physical controls and sensors have been used to support physical interface prototyping [9] or ubiquitous computing environments [2]. Other work has integrated computational devices into tabletop environments. Rekimoto and Saitoh [21] augmented physical objects placed on a table with virtual "auras." Rogers et al. [22] added physical artifacts to a tabletop with the express intention of encouraging collaborative behaviors such as turn-taking. Subramanian et al. [27] observed the importance of "handing-off" physical objects from one user to another in "real-world" tabletop interactions (e.g., constructing a puzzle), and they found that hand-off times for virtual objects were reduced when handing-off a tangible stand-in for the virtual object versus a non-tangible, finger-pointing technique. In contrast, we explore the use of *generic, re-mappable* physical controllers that work in concert with direct-touch controls to support co-located collaborative work.

We share some motivations with other work incorporating generic physical input devices into PC environments. In particular, Hartmann et al. [11] use a physical mixing board to expose control over design parameters, allowing the user to rely on motor control to execute actions. Crider et al. [4] employ a mixing board to control a graphics and visualization system, for reasons of conserving screen display space, improving interaction efficiency, and offering kinesthetic feedback. However, neither work deals with collaboration or touch interfaces.

Finally, also of relevance is work by Forlines et al. [8] and by Müller-Tomfelde and Schremmer [17], who recognized that the standard mouse itself may be a useful input device for multi-touch tabletops in certain circumstances, due to precision and other factors. While we do not use a mouse, we do propose that there are benefits to complementing multi-touch interaction with other input methods.

OUR APPROACH

Based on this existing work, we outline key elements of our design approach for effectively supporting both precise control and collaborative interaction in a tabletop environment for high-precision tasks such as media editing.

- 1. Allow users to choose between direct touch (using finger-based interactions) and indirect control (using a physical controller).** Based on the tradeoffs of direct and indirect control outlined in [10], neither modality may be clearly optimal for a particular task. All editing and system display parameters in our system can be controlled using either modality.
- 2. Provide a single, shared multi-touch display area.** This shared area contains a visual representation of the editing artifact. This central area also contains multi-touch widgets that identify all available editing parameters, visually indicate the current values of these parameters, and expose control over the parameters via touch manipulation. Thus all users have a common display for

planning and observing the editing process, and touch gestures executed on this space are apparent to the group.

3. Provide several small, moveable physical controllers for users to exercise indirect, precise control with tactile feedback. Physical controllers in our system can be distributed to individual users, acting as personal control spaces that supplement the shared group display and control space. This approach supports replication and accommodates territoriality while allowing users freedom over their own positioning around the table.

4. Provide a means for users to map and re-map controllers to editing parameters, via direct touch, and convey the current mapping states visually. Space dictates that individual controllers may not have enough buttons, knobs, or sliders to accommodate all system parameters. Instead, users explicitly decide how to map the continuous and discrete physical controls to a subset of continuous and discrete parameters. The mapping process is executed using direct touch on the shared control space to promote awareness of mapping state across users. Users can re-map controllers at any time. Visual cues indicate mapping state so the user does not have to memorize the mapping state as it changes.

5. Support saving, reloading, and sharing of controller mappings. This may reduce overhead if there is a tendency to reuse mappings over time, and it facilitates collaborative behaviors that are challenging without system support, such as role switching and offline review of group dynamics.

SYSTEM DESIGN AND IMPLEMENTATION

We have implemented the above approach in Ensemble, a new tabletop collaborative audio editing system.

Background: Collaborative Audio Editing

Audio editing is one of several domains requiring precise control over a large number of parameters, and where support for collaborative editing is desirable but poorly supported by existing software. Even audio editing software for novices, such as GarageBand (Apple, Inc.), exposes hundreds of continuous editing parameters, including volume level and pan (the position of the sound between the left and right speakers), and parameters for each of many audio effects such as chorus, distortion, and reverberation. Furthermore, such software typically employs a “track” editing paradigm, where each recorded instrument or audio source may comprise its own unit or track. Though tracks are played simultaneously, each track’s parameters can be manipulated independently, multiplying the effective number of control parameters in the system.

Amateur audio editing seems like a naturally collaborative task; for example, all members of a band are stakeholders in the editing process required to turn the raw output of a recording session into an album. In order to assess the suitability of audio editing as an application domain in which to implement our approach, we conducted a survey

of 51 Microsoft employees who were amateur users of audio editing and mixing software. Survey questions focused on status quo software support for collaboration. Respondents to our survey described many ways that collaboration on audio editing tasks is beneficial, mentioning the creative synergy, the combination of ideas and talents, and “general shared bliss when we both click on an idea.” However, 43 of the 51 respondents (84%) worked alone more often than not, and indicated that this reluctance to collaborate was in part due to the software editing environments: “it’s crowded,” “there is only one mouse and keyboard,” “each person must have their own computer to be effective.” 73% of respondents agreed that they would be interested in tools that made co-located collaboration on audio editing easier.

When asked about potential technological enhancements to make collaboration more satisfying and enjoyable, 94% of respondents agreed that a “big display” would be beneficial, and 76% agreed there was benefit in modifying the interface to allow several users to simultaneously control editing parameters. These properties match well with the affordances of tabletop systems. In fact, several tabletop-like systems have been created for music performance and audio synthesis (but not editing), including Audiopad [20] and Reactable [14], both of which employ tangible tags as input devices, and a multi-touch system by Davidson and Han [6] that does not integrate physical artifacts.

While Ensemble seeks to promote musical collaboration in a similar spirit to these systems, the task domain is quite different (editing existing audio rather than synthesizing or performing), and our use of physical artifacts serves to complement multi-touch interactions by providing needed feedback and precision. In this, our work also relates to a larger context of research regarding the design of appropriate physical and virtual interfaces for musical performance and audio control (e.g., [12,30]).

In addition to the design goals outlined in the previous section, Ensemble was designed around a goal of being usable by people with no previous audio editing experience, so we exposed fewer and simpler control parameters than might appear in a commercial editing package.

Ensemble

The Ensemble system was built on a prototype interactive tabletop (see Figure 1) that employs an under-table infrared vision system to track fingers and objects on the table’s surface. Two adjacent overhead projectors provide the display, yielding an overall display resolution of 1024x1536. The tabletop itself measures approximately 120cm wide by 180cm long, and stands 90cm high. The software was written in C# and WPF and uses a version of Microsoft’s Surface SDK that was modified for this table’s vision system.

Central Editor

The central editor object, shown in Figure 2, presents a visual representation of audio tracks in a manner similar to

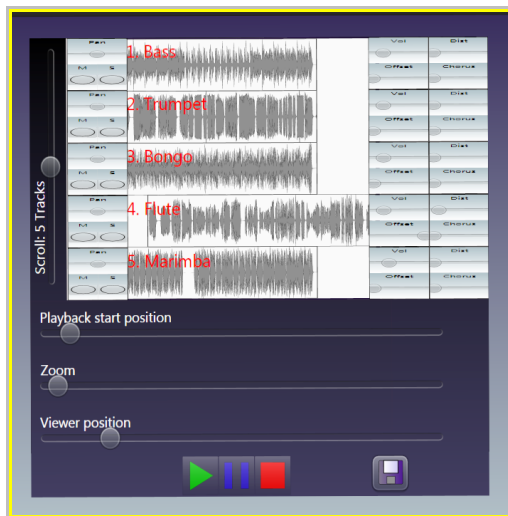


Figure 3: The central track editor. All parameters can be interacted with directly via direct-touch manipulations, or can be mapped to physical controls.

commercial editing software. This space provides touch controls for all audio editing and display parameters. Each track includes a waveform representation of the corresponding audio, and controls for seven standard audio editing parameters appear alongside each waveform. The two discrete, on-or-off parameters are mute (disable this track) and solo (disable all tracks other than this one). The five continuous parameters are volume, pan, offset (the point in time when a track begins playing, relative to the other tracks), level of “chorus” effect, and level of “distortion” effect¹. The two discrete parameters are controlled using touch buttons, and the five continuous parameters are controlled using horizontal touch sliders. These touch widgets are sized to be approximately finger-sized by default to facilitate easy manipulation, though users are able to resize the editor to make the display and widgets larger or smaller. The central editor also contains buttons to play, pause, and stop the audio, as well as slider controls for vertically scrolling to select the tracks currently displayed, zooming in and out on the track waveforms, horizontally scrolling the waveforms, and changing the playback position (i.e., the starting position of the audio when the play button is pressed). Users can control the zoom and horizontal viewing position using direct-touch manipulations on the track waveforms, and they can control the size, position, and orientation of the editor itself using multi-touch gestures on the editor border.

Physical Controllers

Four physical controller boxes are integrated into the

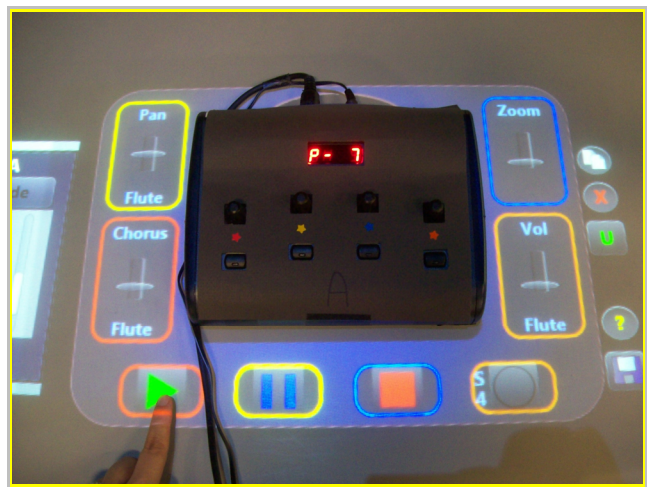
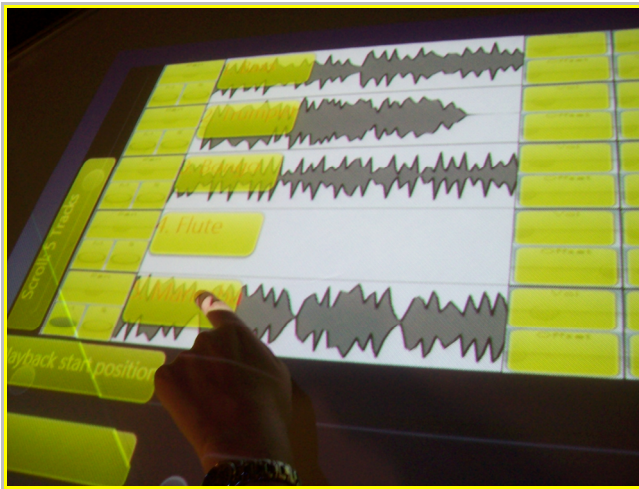


Figure 2: Physical controller with virtual, multi-touch aura. The aura contains eight “slots.” The four slots below the box contain *virtual buttons*, which are mapped to the same discrete editing parameters as the four *physical buttons* on the box. The four slots on the sides of the box contain *virtual sliders*, which are mapped to the same continuous editing parameters as the four *physical knobs* on the box.

system. Each box (Figure 3) is a Nano BCN44 controller (Behringer Intl, GmbH) containing four continuously rotating knobs and four buttons. Each box communicates with the tabletop software to indicate when each knob is incrementally turned to the right or left, and when each button is pressed and released. This particular controller was chosen for its size (with an 22cm by 17cm footprint, several can fit on the tabletop with space to spare, see Figure 1), programmability, low cost, and accommodation of both continuous and discrete controls.

On the bottom of each box is a tag that uniquely identifies the box to the table’s vision system, allowing a visible aura to track the box as it moves around the table. Inside this aura are several touch-sensitive buttons used to manage the control mapping state, as well as eight “slots,” each of which corresponds to one knob or button, the identity of which is determined by a simple color-coding scheme (Figure 3). When a knob or button is mapped to a parameter, the corresponding slot will be filled with a graphical representation of that parameter and a uniquely identifying label (the track number and/or control name). Whenever an editing parameter’s value changes, regardless of how the change was initiated, all graphical representations of that control will change in synchrony. For example, if a user manipulates the Flute Chorus control on the central editor, the slider in Figure 3 will simultaneously update to reflect the change. Or, if a user manipulates the Flute Volume using the knob on the controller in Figure 3, the adjacent widget and the Flute Volume slider control on the central editor will similarly change. Virtual widgets displayed in a physical controller’s aura can be controlled by direct touch. Both virtual controls and physical knobs are mapped linearly to the editing parameters, which were bounded by parameter-specific

¹ “Chorus” and “distortion” are common effects used in music production. Increasing the level of chorus applied to an audio track gives the impression that the track is being played by an increasing number of sound sources. Increasing the level of distortion applied to an audio track makes the track sound “rougher” and “fuzzier”. These effects were selected because they are both common in audio processing and easily perceived by novices.



(a)



(b)

Figure 4. Assigning parameters to a physical controller box. (a) The central editor in “copy” mode. Touching any of the highlighted parameters will place that parameter in the user’s clipboard. (b) The controller box that initiated “copy” mode is surrounded by a yellow halo. Touching a continuous control in the clipboard above the box has activated the virtual “slots” (now highlighted in green) corresponding to the box’s four physical knobs. When any of these slots is touched, the selected control will be mapped to the corresponding knob.

ranges (e.g., 0 to 1 for volume, -1 to 1 for pan). A 360-degree turn of a knob corresponded to approximately the entire range of a parameter (or to 10 seconds for offset).

Parameter Mapping Process

When Ensemble initializes, none of the knobs and buttons are mapped to control parameters, and their corresponding aura slots are blank. Users can construct a mapping from knobs and buttons to parameters using the following process, which is summarized in Figure 4.

First, pressing a controller’s touch-sensitive “Copy” toggle button, located on the right side of the aura for each box, initiates the process of mapping (i.e., “copying”) editing parameters to that controller. This causes a “clipboard” space to appear above the controller aura, and a yellow highlight to appear around the aura’s border. This also causes the central editor to display bright yellow virtual touch buttons around all of the available controls, including track controls, playback controls, and editor controls such as zoom (Figure 4a). Second, touching any of these yellow buttons causes a copy of the corresponding control widget (a button or slider) to appear in the controller’s clipboard (Figure 4b, top). Third, touching a control in the clipboard selects it, causing green highlights to appear over the aura slots corresponding either to knobs (if the selected control represents a continuous parameter) or to buttons (if the selected control represents a discrete parameter). Finally, touching one of these green highlights will cause the control to leave the clipboard, map the associated knob or button to control the parameter, and populate the slot with the corresponding (labeled and touch-enabled) widget.

The user can touch the “Copy” button once more to end the mapping process for that controller and return the central editor to its normal state. Or, touching the “Copy” button for another controller box will transfer control over the

mapping process to that box, causing the yellow highlight to transfer to the new box’s aura, and causing controls copied from the central editor to now appear in the new box’s clipboard. At any time, only one controller can control the mapping process, as indicated by possession of the yellow aura highlight.

Ensemble does not prohibit multiple users from copying the same parameters to their controllers, nor impose any assumptions regarding which parameters should be mapped to which controllers. Users can modify the mapping of a controller box at any time. Touch buttons on the aura allow the user to delete any mapping and undo any mapping deletion or creation. Additionally, users may press “Copy” again at any time to reopen the clipboard and initiate another mapping session to that controller.

Ensemble supports saving the mapping state of any controller box—that is, which knobs and buttons are mapped to which editor parameters. When a user presses the “Save” button in a controller box’s aura, the current state is saved and appended to a local, touch-scrollable history list, from which the user may reload any saved mapping state to the box. Saved states may also be loaded into another control box by dragging and “docking” the history list to the chosen box.

EVALUATION

Our approach raises several questions regarding its impact on the collaborative user experience. For example, do users actually employ the physical controllers to achieve precise control and create an individual, local control space? Given that users have several options for exercising control (local physical controllers, local touch controls, central touch controls), are any of these options clearly preferred? Are the collaborative benefits of shared multi-touch tabletops maintained despite some interactions being performed using

indirect control? Is the mapping process useful, and what implications does it have for the collaborative experience?

To investigate these questions, we conducted an observational study of groups of users collaborating on audio editing tasks using Ensemble. Our observations included video footage, automatic logging of all user actions by the system, and written questionnaires.

Participants

We recruited 40 Microsoft employees to form 10 groups of 4 participants each. Due to cancellations, the 10 groups varied in size: 6 groups had 4 members, 2 groups had 3 members, and 2 groups had 2 members, for a total of 34 participants (3 female). Participants' ages ranged from 20 to "over 55." 29 subjects had used multi-touch technology "once or twice," 5 had never used multi-touch technology, and none had used multi-touch on a regular basis. 24 participants self-identified as musicians, and 29 reported having used audio editing software previously.

Task

User groups consisted of people who were not necessarily musicians, who did not necessarily have significant audio editing experience, and who did not know each other or necessarily share subjective opinions about audio content. Therefore, the group editing task was selected to be both objective and accessible to novices. Specifically, each group was asked to perform three "matching" tasks. In each task, the group was presented with a set of four or five unedited audio tracks along with an audio prompt lasting between 15 seconds and 1 minute. In a real-world audio editing scenario, participants would work together to edit the raw tracks to meet a shared set of subjective, aesthetic criteria; here, participants were asked to work as a group to edit the raw tracks to sound like the prompt. The prompts were created by one experimenter prior to the experiment using Ensemble's physical and virtual controls, starting from the same audio tracks provided to the group. Prompts were designed so that each task could be completed in around 10 minutes (which we verified with pilot experiments). The three prompts used in our experiment are available at <http://research.microsoft.com/~dan/ensemble>. Each task required the editing of a somewhat different set of audio parameters. Participants used the Ensemble system as described earlier, but augmented with a timer showing the time remaining in the task, and virtual buttons for playing, pausing, and stopping the prompt.

Each group was given a 5-minute hands-on tutorial on multi-touch tabletop manipulations, followed by a 15-minute tutorial on the functionality of Ensemble and the matching tasks. Groups were given 10 minutes to match each of the 3 prompts, for 30 minutes of total task time. The order of the prompts was varied among groups as a Latin square with an incomplete repetition. At the conclusion of the study, individual participants completed questionnaires.

RESULTS

Effectiveness of System and Task

The participants understood the system well, and the matching task presented them with a challenging but achievable assignment. Of the 34 participants, 97% agreed that they understood how to use the system (rating 4 or 5 on a 5-point Likert-style scale; median score of 5).

Unlike a typical matching task study, we were not interested in correlating the degree of success with aspects of the task or user behaviors. In fact, this degree is hard to measure objectively, since simple metrics such as Euclidean distance in parameter space do not correlate well to perceptual similarity for these parameters. Rather, we were concerned that the task focused the participants on a common, reasonable goal, so that we could explore participants' use of the integrated physical controllers. 82% of participants either somewhat agreed or strongly agreed that they were able to successfully perform the matching tasks (median Likert of 4). Given a rough objective "success" criterion of the percentage of parameters that were edited to within 20% of the prompt value, groups succeeded for a median of 75% of the available parameters on each prompt (minimum 44%, maximum 92%).

Supporting Precision

The physical controllers were perceived to be helpful in supporting precision manipulations, and the controllers were perceived to provide a greater accuracy advantage for continuous parameters than discrete parameters.

A Friedman test comparing participants' 5-point Likert scale ratings of perceived accuracy for discrete and continuous parameters using either physical or touch controls found significant differences ($\chi^2(3, N=23) = 49.77, p < .001$) (see Table 1). Follow-up pairwise Wilcoxon tests found that all pairwise differences were also significant. These tests showed that participants felt that the physical controls were significantly more accurate than the touch controls, for both continuous parameters ($z = -5.11, p < .001$) and discrete parameters ($z = -3.37, p = .001$). Tests also showed that participants felt that the physical controls provided a significantly greater accuracy boost for the continuous parameters than the discrete ones ($z = -2.12, p = .03$), whereas they felt that the touch controls were more accurate when dealing with discrete, rather than continuous, parameters ($z = -3.81, p < .001$).

The Collaboration Experience

Our observations suggest that our system is effective at supporting equitable participation and individuals' awareness of the group's activities. 85% of participants strongly or somewhat agreed that everyone in their group participated

Control Type	Median	Mean	Std. Dev.
physical, continuous	5	4.79	0.59
touch, continuous	2	2.15	0.86
physical, discrete	5	4.50	0.83
touch, discrete	3	3.15	1.06

Table 1: Five-point Likert ratings for perceived accuracy; differences between all control types are significant ($p \leq .03$).

equally. Our system was designed to support cooperative work by providing awareness of group members' actions, despite the introduction of individual control devices, through the preservation of the shared central editor and through mechanisms such as the ability to query the location of individual controls. This awareness support succeeded: 85% of participants strongly or somewhat agreed that they were aware of others' editing actions, and 94% strongly or somewhat agreed that their level of awareness of others' actions was adequate.

Establishment of Individual Control Spaces

Results show that participants used the controller boxes to establish personal control spaces. During the introductory tutorial, groups were instructed that the controller boxes could be used as they wished, to be distributed to individuals, used collectively by the group, or not used at all. One group chose not to use controller boxes at all for their second prompt. Otherwise, members of four-person groups each used one box per person exclusively throughout the task (and used the same box across prompts). Members of both two-person groups and one three-person group either used one box exclusively throughout the task, or used one box until they ran out of empty slots, then switched to an "extra" (unassigned) box. In the other three-person group, one very dominant person controlled the conversation and executed the majority of the edits; after the first prompt, the other two users gave up and did not create any mappings for their boxes. Even in this case, however, they stood behind the box they had used for the first prompt. The boxes were never used collectively or shared among users. 23 participants (68%) strongly agreed and 8 (24%) somewhat agreed that they treated one controller box as their own. In short, the controller boxes were consistently used as individual control spaces, and any extra boxes either sat unused or were eventually assigned to individuals who ran out of room for new controls.

The tracking mechanism of the table allowed users, in principle, to bring their control space (the box and its aura) with them when they changed position to better see or reach the central editor, or communicate with another user. We did not observe this behavior, however. The maximum distance a controller box ever traveled from its starting position over any 10-minute task was about 75cm, and the vast majority moved less than 30cm; considering the 2.2-square-meter size of the table, these movements are quite small. Two users, from different groups, often walked up and down the table in order to touch or gesture at the central editor, but most users stayed in place, directly behind their controllers, and resorted to reaching (sometimes far) or asking others nearer the central editor to execute a touch manipulation on their behalf. It is unclear whether this is because the users felt territorial claims to particular parts of the table as suggested by Scott et al. [24], not just to the controller, or because they found moving the controllers cumbersome (the controllers were not wireless, but their mobility was enhanced by long extension cords). In any case, our observations do not support the

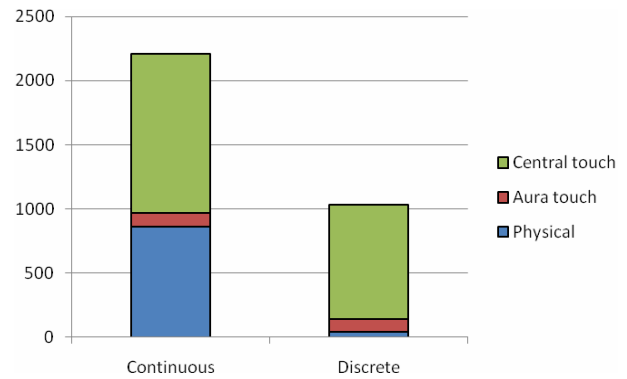


Figure 5: The total number of edits for all tasks, by parameter type and manipulation type.

conclusion that the *mobility* of personal control spaces effectively addresses reach or orientation issues.

Physical vs. Virtual Controls

25 participants (74%) said they preferred physical control overall. Of these, 16 mentioned accuracy or responsiveness and 6 mentioned feedback when answering a free-response question asking them to explain their preference. Only six participants (15%) strictly preferred touch controls, citing reasons such as intuitiveness, the presence of visual feedback and an identifying context in the exact same location as the gesture, and ease-of-access (in that no mapping was required to access the central touch controls).

Despite preferring physical controls overall, participants made extensive use of touch controls. **Figure 5** shows the total number of edits made using each type of control, for discrete and continuous parameters. (These numbers are aggregated over all users due to our table's lack of ability to differentiate touch identity for edits made using the central editor.) For all tasks, the number of touch edits was high: 78 per prompt on average, as compared to 30 edits using the physical controllers. The difference in behavior between continuous and discrete parameters was quite pronounced, with 39% of all continuous parameter edits executed via the physical knobs, and just 4% of all discrete parameter edits executed via the physical buttons. Considering only the track parameters, which were more frequently mapped to the control boxes, users employed the physical controls roughly as often as touch controls (either central touch controls or those on the aura) for continuous parameters, and used touch controls more often than physical controls for discrete parameters.

Users were less likely to map discrete parameters than continuous controls to physical controllers (only 109 of the 475 controls ever mapped were discrete). When discrete parameters were mapped, users employed the touch controls on the local aura more often than the buttons on the controller itself (in total, 99 edits used touch, 40 used physical buttons). When asked for reasons why they employed local touch buttons instead of physical buttons, despite participants' perception that physical buttons were more accurate,

common explanations were the increased visibility and “ease of use” of the local touch buttons.

Central vs. Local Controls

68% of participants stated a preference for using local controls (physical and touch), and 32% preferred using the central controls. “Reach” was cited as a reason for preferring both locations: local controls were easier to reach for most people, but people positioned near the editor could reach a greater number of controls easily. People who liked central controls mentioned their greater visibility to the group and the fact that the controls were surrounded by more contextual information. Often, users employed the central editor even when they realized they might be sacrificing precision or had to stretch a great distance to reach it, citing that it was easy to use for rough or one-time edits, without incurring the overhead of mapping. The central editor also served an important purpose beyond offering touch control; as a shared, visible space, it was often the object of pointing gestures, and employed to focus group conversation and attention.

Mapping and User Roles

With the majority of users expressing ownership over one box, the process of mapping the controllers to parameters can be considered in terms of mapping the *users* to parameters. This notion was reflected in that users referred to the mapping as a process of establishing ownership, such as, “Why don’t you take the volume?” or “I’ve got the guitar controls.” The mapping process was used as a tool for users to delegate or assert responsibility in the group. For some groups and tasks, users verbally agreed on a strategy for delegating parameters before anyone began to copy parameters. Other times, one assertive user would assign roles to the others (or to himself) without consensus.

Group work often iterated between a planning phase, in which participants delegated responsibility for and set up mappings to a group of parameters they anticipated editing soon, and actually performing edits. “One-off” edits that were not planned in advance were more often made using the central editor; such edits took place outside the role structure established by the planning phase.

We did not observe participants intentionally assigning the same parameters to multiple people; of the 496 instances in which parameters were mapped to controllers, only 24 of these duplicated parameters that were already mapped elsewhere. Throughout each task, participants also tended to verbally establish that no other group member possessed a parameter before mapping it to their own controller.

Users sometimes subverted the cooperative process by hitting “Copy” and selecting parameters to add to their clipboard, without announcing this to the group. When more than one user acted independently in this way, without others being aware of their actions, this resulted in confusion, as users ended up “stealing” the copy privilege from each other. Five participants noted that the tight cooperation that was necessary to establish mappings and/or user roles was the most challenging aspect of collaboration with the system.

These observations suggest that forcing users to create the mappings from scratch, and to take turns establishing mappings rather than acting simultaneously, encourages explicit communication regarding delegation of responsibility. This process may be one reason there was such high awareness of other users’ mappings and actions, though it may involve a greater degree of explicit planning than some users prefer.

The abilities to save, reload, and share mappings were not exercised by users in these tasks, and participants neither agreed nor disagreed that they were useful in this context (median Likert score 3 out of 5). Four participants annotated their Likert ratings, unprompted, to add that such functionality would be appreciated in longer or more complex tasks. Given the tight relationship between user role and controller mapping, it seems reasonable that these functions and others like them could contribute to promoting rich and dynamic user roles.

Mapping and Task Strategy

We observed users employing the mapping to change categories of user responsibilities according not only to the group style (e.g., cooperative turn-takers, or passive and active participants) but also to the perceived task requirements. For example, for Prompt B and Prompt C, 8 and 9 groups, respectively, assigned individuals control over the parameters of a particular track. For Prompt A, on the other hand, the first thing most groups noticed when listening to the prompt was that the sound events proceeded in a strict timing sequence, and 3 groups responded by initially assigning all offset controls to a single user, who could then recreate the timing sequence. For this prompt, only 4 groups employed the strategy of dividing the work by tracks, and the 2 other groups that used the controllers at all employed different strategies. In most tasks, at least one person modified his or her mapping from its initial state in a way that was not necessarily consistent with the original strategy (i.e., not necessarily adding another parameter for the same track, or adding another parameter of the same type as those already mapped). The variation of mapping among groups, users, and tasks underscores the need to support flexible and changeable mappings.

DISCUSSION

By observing 10 groups use our dynamically re-mappable controllers in the context of the Ensemble audio editing system, we observed several consistent themes in users’ interaction with and perception of the system.

Physical sliders were perceived to be more accurate than virtual sliders, and physical buttons were perceived to be more accurate than virtual buttons, but the difference in perceived accuracy was less pronounced for buttons. Most users stated they preferred physical manipulations to touch manipulations, citing reasons including precision and feedback. However, some users preferred touch manipulations, for reasons such as the increased visual context. Overall, users overwhelmingly chose to control locally mapped sliders using physical knobs rather than the

touch sliders on the aura, but they used the aura touch buttons slightly more often than physical buttons for the locally mapped discrete controls. Our approach's mixture of touch and physical control options for each parameter accommodated these varied preferences.

Most users stated they preferred local controls to central controls. However, the central, shared control space remained a focal point of interaction for all groups. Users often gestured or pointed at the central editor when discussing the task. Many touch manipulations were performed on the central editor even in groups that heavily utilized the physical controllers, due to increased contextual information, visibility, and immediacy; the central editor was especially used to make quick and rough edits.

Users had a strong tendency to claim ownership over one controller and over the parameters to which it was mapped. The controllers became individual control spaces that reflected the roles of the users.

Users exhibited different strategies for mapping the controllers, and strategies varied across both task and group, suggesting that a static controller mapping would be inadequate. Furthermore, the mapping strategy was directly related to the roles of users in the group, and was used to delegate or assert ownership over particular controls. Multiple users trying to control the mapping process at the same time, without communicating their intentions to each other, occasionally created confusion.

Beyond Ensemble

Our approach of integrating dynamically re-mappable controllers into tabletop group editing system can be applied in a variety of collaborative, precision-sensitive editing tasks, such as graphic design, print or Web layout, scientific visualization, or video editing. It is sensible that controller hardware and mapping strategies might vary for other domains, as our use of knobs mapped to one-dimensional parameters was motivated by existing audio hardware interfaces. The knobs might be replaced by other controls that are appropriate for a different task and/or already familiar to domain experts. For example, a joystick may be appropriate for 2D positioning tasks. The granularity of the mapping from hardware to parameters will also be application- (and possibly parameter-) dependent, according to the necessary precision and range. The integration of a different controller would require specifying the number and type (e.g., continuous, discrete) of the new inputs, the relation mapping each new input to application parameters, and the visual appearance of the controller's aura. Infrastructure similar to the Input Configurator toolkit [7] could facilitate the dynamic specification of more complex mappings by the users, when hard-coded mappings are insufficient.

The appropriateness and appearance of our local aura visualization method also depends on the choice of controller, the size and type tabletop, and the nature of the application GUI. In the event of a small tabletop, or an

application that must not be visually occluded, there must be another way to visually inform users of the mapping state of a controller. Controllers with built-in reprogrammable LCD indicators, or a top-projection system projecting onto controllers with fixed positions off of the table might offer such feedback. On the other hand, the occlusion presented by small, relatively stationary controllers may be of less consequence than that of many arms and hands when only touch is used in a smaller collaborative tabletop environment.

We expect the nature of the interaction with any system to depart from our observations for tasks involving very few parameters (especially if all parameters can be mapped to one or two controllers simultaneously). In this case, there may be less incentive for users to divide responsibility amongst themselves, and other infrastructure may be more appropriate to support broad participation. On the other hand, we anticipate that our approach will scale well to tasks where there are many more parameters.

Summary of Key Findings and Proposed Guidelines

Reflecting on our experiences observing groups using Ensemble, we highlight several phenomena that informed our own evaluation of the system and suggest how such an approach might be most useful in future applications:

- Ensemble's physical controllers became de facto personal control spaces. Therefore, we propose that providing one controller per user is a reasonable configuration for collaborative systems of this nature.
- Pilot subjects expected the local aura to be controllable by touch, and study participants often used touch control on the aura, especially for discrete parameters. Therefore, we propose that local virtual representations of discrete controls be touch controllable if possible.
- Participants used physical controllers for parameters that needed to be manipulated very precisely, parameters that they expected to need more than occasionally, and logical groups of parameters that they could assign to an individual participant (such as controls for a track). While participants exercised the ability to map both continuous and discrete parameters into a local control area, they found the physical controller less essential for discrete parameters, and often opted for local virtual buttons. We therefore understand the usefulness of re-mappable physical controllers to be highly dependent on the types of parameters present.
- We observed that the shared multi-touch control area still offered practical benefits (such as making quick edits without incurring the overhead of mapping) and collaborative benefits (such as pointing at the space when talking, or making changes that are visible to the group), even when precision of some touch manipulations was inadequate. This suggests that touch interaction and physical controllers play complementary roles.
- Based on our experiences, the choice of mapping creation paradigm may support certain group collaboration styles and discourage others. A paradigm such as ours that

serializes the mapping process across users might be desirable if group awareness and communication are critical. Alternatively, greater efficiency may be achieved by allowing users to create mappings in parallel, which would be feasible if our approach is combined with an identity-differentiating tabletop technology such as DiamondTouch [5].

- Given the observed relationships between user roles and controller mappings, the set of allowable mappings may strongly influence the types of user roles that are allowable. Designers might restrict the mappings to try to enforce more equitable participation (everyone must have the same number of controls), turn taking (only one person's controls are active at any time), or other work styles. We hypothesize that supplying users with shortcuts to mapping presets might also bias the delegation of responsibilities along lines suggested by the presets. Without restrictions, users can employ a variety of mapping strategies based on their own criteria.

CONCLUSION

In designing a collaborative system integrating physical controls with a multi-touch tabletop, we hoped to support both collaboration and precision. Our observational study of Ensemble, our prototype collaborative audio-editing system, suggests that physical controllers can complement touch-based interactions and support a productive collaborative experience in a domain not typically served by co-located collaborative software.

ACKNOWLEDGEMENTS

We thank Björn Hartmann and Hrvoje Benko for their help.

REFERENCES

1. Albinsson, P., and Zhai, S. High Precision Touch Screen Interaction. *CHI 2003*, 105-12.
2. Ballagas, R., Ringel, M., Stone, M., and Borchers, J. iStuff: A Physical User Interface Toolkit for Ubiquitous Computing Environments. *CHI 2003*, 537-44.
3. Benko, H., Wilson, A., and Baudisch, P. Precise Selection Techniques for Multi-Touch Screens. *CHI 2006*, 1263-72.
4. Crider, M., Bergner, S., Smyth, T., Moeller, T., Tory, M., Kirkpatrick, A., and Weiskopf, D. A Mixing Board Interface for Graphics and Visualization Applications. *Graphics Interface 2007*, 87-94.
5. Dietz, P., and Leigh, D. DiamondTouch: A Multi-User Touch Technology. *UIST 2001*, 219-26.
6. Davidson, P., and Han, J. Synthesis and Control on Large Scale Multi-Touch Sensing Displays. *New Interfaces for Musical Expression 2006*, 216-9.
7. Dragicevic, P., and Fekete, J.-D. The Input Configurator toolkit: Towards high input adaptability in interactive applications. *Advanced Visual Interfaces 2004*, 244-7.
8. Forlines, C., Wigdor, D., Shen, C., and Balakrishnan, R. Direct-Touch vs. Mouse Input for Tabletop Displays. *CHI 2007*, 647-56.
9. Greenberg, S., Fitchett, C. Phidgets: Easy Development of Physical Interfaces Through Physical Widgets. *UIST 2001*, 209-18.
10. Ha, V., Inkpen, K., Mandryk, R., and Whalen, T. Direct Intentions: The Effects of Input Devices On Collaboration Around a Tabletop Display. *IEEE Tabletop 2006*, 177-84.
11. Hartmann, B., Yu, L., Allison, A., Yang, Y., and Klemmer, S. Design as Exploration: Creating Interface Alternatives through Parallel Authoring and Runtime Tuning. *UIST 2008*, 91-100.
12. Hunt, A., and Kirk, R. Radical User Interfaces for Real-time Control. *IEEE EUROMICRO 1999*, 2006-12.
13. Ishii, H., and Ullmer, B. Tangible Bits: Towards Seamless Interfaces Between People, Bits, and Atoms. *CHI 1997*, 234-41.
14. Jorda, S., Kaltenbrunner, M., Geiger, G., and Bencina, R. The ReacTable*. *Intl Computer Music Conf 2005*, 579-82.
15. Microsoft Surface, <http://www.microsoft.com/surface/>
16. Morris, M., Paepcke, A., Winograd, T., and Stamberger, J. TeamTag: Exploring Centralized versus Replicated Controls for Co-located Tabletop Groupware. *CHI 2006*, 1273-82.
17. Müller-Tomfelde, C., Schremmer, C. Touchers and Mousers: Commonalities and Differences in Co-Located Collaboration with Multiple Input Devices. *CHI 2008*, 1149-52.
18. Olwal, A., and Feinter, S. Rubbing the Fisheye: Precise Touch-screen Interaction with Gestures and Fisheye Views. *Conf. Supp. UIST 2003*, 83-4.
19. Pangaro, G., Maynes-Aminzade, D., and Ishii, H. The Actuated Workbench: Computer-Controlled Actuation in Tabletop Tangible Interfaces. *UIST 2002*, 181-90.
20. Patten, J., Recht, B., and Ishii, H. Audiopad: A tag-based interface for musical performance. *New Interfaces for Musical Expression 2002*.
21. Rekimoto, J., and Saitoh, M. Augmented Surfaces: A Spatially Continuous Work Space for Hybrid Computing Environments. *CHI 1999*, 378-85.
22. Rogers, Y., Lim, Y.-K., Hazlewood, W. Extending Tabletops to Support Flexible Collaborative Interactions. *IEEE TableTop 2006*, 71-8.
23. Ryall, K., Morris, M.R., Everitt, K., Forlines, C., and Shen, C. Experiences with and Observations of Direct-Touch Tabletops. *IEEE TableTop 2006*, 89-96.
24. Scott, S. D., Sheelagh, M., Carpendale, T., and Inkpen, K. M. Territoriality in collaborative tabletop workspaces. *CSCW 2004*, 294-303.
25. Shen, C., Vernier, F., Forlines, C., and Ringel, M. DiamondSpin: An Extensible Toolkit for Around-the-Table Interaction. *CHI 2004*, 167-74.
26. Stewart, J., Bederson, B., and Druin, A. Single Display Groupware: A Model for Co-present Collaboration. *CHI 1999*, 286-93.
27. Subramanian, S., Pinelle, D., Korst, J., and Bull, V. Tabletop Collaboration through Tangible Interactions. *IEEE WETICE 2007*, 412-17.
28. Ullmer, B., and Ishii, H. The metaDESK: Models and Prototypes for Tangible User Interfaces. *UIST 1997*, 223-32.
29. Vogel, D., and Baudisch, P. Shift: A Technique for Operating Pen-Based Interfaces Using Touch. *CHI 2007*, 657-66.
30. Wanderley, M., and Orio, N. Evaluation of Input Devices for Musical Expression: Borrowing Tools from HCI. *Computer Music Journal 26(3)*: 62-76, 2002.
31. Wu, M., Shen, C., Ryall, K., Forlines, C., and Balakrishnan, R. Gesture Registration, Relaxation, and Reuse for Multi-Point Direct-Touch Surfaces. *IEEE TableTop 2006*, 183-190.