Macbett: A Case Study of Performance & Technology for Dynamic Theater Spaces
Jeff Burke, Fabian Wagmister, Adam Shive
HyperMedia Studio, School of Theater, Film and Television
University of California, Los Angeles
{jeff, fabian, adam}@hypermedia.ucla.edu

Abstract

This paper describes the technical and artistic development of a mainstage subscription-series production of Eugene Ionesco’s Macbett that incorporated basic interactive systems to enable real-time relationships between onstage action and design elements. This research and performance involved students in Theater, Electrical Engineering, Music, Film and Television, and Computer Science. Sensing and production systems included commercial performer tracking, large-scale DMX lighting, sound analysis, processing, and playback, all integrated by software components developed for Windows and MacOS. Individual systems communicated using UDP over an ethernet network. A brief motivation, technical overview, and examples of interactive sequences from Macbett are given.

Introduction

The desire of theater directors and designers to create an evocative experience for their audience has driven modern theater technology for lighting, projection, sound, and scenery to provide amazing capabilities for both expressive subtlety and sophisticated spectacle. Yet unanswered by this technology has been the definitive shift in media and performance away from traditional boundaries and towards interactive experiences capable of responding to viewer and performer. Each new advance in theatrical technology is applied to a fixed structure of control, where the relationship between performers and their environment is simulated by a static set of cues built around a small set of technical rehearsals; what was once a technological necessity has become a design paradigm. The more intricate the technical sequence, the less it can be adjusted during performance and the more constrained the actors’ performances must become. The modern computerized approach—cues pre-programmed with static timing—need no longer be the only way to create complex stage design.

Musicians, dancers and choreographers were some of the first to explore interactive technology that connected performers (and often viewer-participants) with digitally-controlled media in their environment. They have driven technical and aesthetic growth in digital art for many years, using new technology in “traditional” performances, creating new genres of performance using digital media, and offering critical comment on technology in their art and writings. Performance artists have likewise embraced and criticized technology in their work. [2,3,20,23] Though some experimental theater has incorporated similar systems, they have rarely been extended into “traditional” theater. [2,4,10,21,23] In the United States, Mark Reaney’s work at the University of Kansas [18] and David Saltz’s at the University of Georgia [21] are examples of larger productions using digital technology as more than just a new source of projection. These are fairly unique; theater in its most familiar form, with a group of actors, a text, lights, sound, perhaps projection and scenery, and a defined playing and audience space, has remained largely on the periphery of new cross-disciplinary art arising out of “the digital.” [10]

Yet many, like Schechner, [22] argue against the disappearance of live theater as a result of new media and information technology. Theater is an ancient art that has not only “survived” but incorporated gas and electric lighting [19], every sort of machinery [7], projected images [26], and amplified sound. [11] The references in texts like Howard’s Bibliography of Theatre Technology [7] provide ample evidence of theater’s long-standing inclusion (and challenge) of new techniques and machines. This encourages us to explore not only new genres arising out of the unique possibilities of digital technology, but its application to traditional modes of performance.
The use of technology by a few important modern directors in the West is discussed briefly as research motivation by Sparacino in [24]. Kotwal provides a more extensive discussion using theater artists Brecht, Craig, Appia, Piscator, Svoboda, and Grotowski as examples. [10] (Grotowski's own Towards a Poor Theatre [5] is a useful contrast in its criticism of all technological spectacle in theater.)

This paper describes our recent research in one particular area: systems that allow designers to experiment with connections between performer action and design in a large production setting. The approach has much in common with the interactive stage and story environment research already taking place, with an important distinction. [17,24,25,28] It does not attempt to produce stage systems that understand what the actors are doing, follow a script based on their movement and speech, or provide improvisational media response to human action. We accept simpler capabilities in return for reliability and very few limitations on production design. As with the work of Saltz [21] and Lovell [13], it depends on human directors and designers to author relationships between onstage action and design elements. In the useful taxonomy of “computer theater” in [16], these are computerized stages.

Many of these intelligent stage systems rely on computer vision techniques that do not perform well under the drastic lighting shifts typical in live performance, preventing (for now) their extension into traditional performances. Here, we used an ultrasonic performer tracking system that requires no such lighting limitation. It also provides consistent identification of performers independent of their costume and makeup, which is difficult or impossible to achieve with current computer vision technology, radar, or in-floor pressure sensors. The major disadvantage is that performers must wear a tracker—a wireless microphone with an element that cannot be covered or obscured. Tracking human movement as a point in space is also limiting; it provides no gesture information like that available from computer vision systems.

**Technology**

As stage design becomes more complex and includes a variety of media, it becomes more difficult to maintain the control flexibility once possible with manually operated sound and lighting systems. Though computer-controlled equipment can handle many hundreds of static cues and perform incredibly complex cross-fades between them, they do so automatically according to pre-determined timing information. Projection, live video, and motorized scenery tend to use even more rigid control structures; in the development of more complex theater systems, repeatability and rehearsal editing features have taken precedence over flexible run-time control. [1] It is difficult or impossible for an operator or designer to safely “reach into” a cue sequence during a performance and adjust parameters of the cue while it is active. Adjustments are instead most often made by having a person decide when to stop and start pre-timed cues. Performers’ actions do drive the progression through most of the lighting and sound design. When performers reach certain points in the script or in their movement, cues are started by the stage manager that adjust lighting, sound, or other stage elements in accordance with a predetermined design. [6,9] More subtly, if the performers’ pacing seems too slow or too fast, the stage manager might call these cues at a different tempo to try to bring the show back to the “right” pace. [6]

The surprisingly extensive technical infrastructure of modern theaters, which enables the creation of these amazingly complex cues, can be connected to sensing systems to create direct relationships between onstage action and the many parameters of digitally controlled or produced design elements. A pragmatic benefit of such a system is that actors would not need to “hit their mark” exactly or cross the stage at the right speed in order to be properly lit. Simple modifications to lighting based on performer position could make complex cues more flexible and responsive to performers. Rather than estimating the time it takes a performer to cross the stage in rehearsal and programming a fade time based on that estimate, instruments could be dimmed directly in performance based on the actor’s position.

---

1 Huntington [8] and Walne [27] provide technical overviews of modern theater control and effects systems.
Clearly, not all relationships between action and design in performance can be expressed spatially, but this is a practical starting point for experimentation. Once a system has been put in place to gather performance data and control production equipment based on it, more sophisticated and unconventional relationships between performers and the stage environment become possible. For example, sound could be intensified based on the speed of movement of an actor during a certain section or lighting controlled by the distance from one tracked performer to another, regardless of where they are on stage.

We developed a suite of tools to explore these possibilities onstage in a traditional production setting. They were applied in the UCLA Department of Theater's production of Ionesco's *Macbett* in June 2001, which was directed by M.F.A. candidate Adam Shive.

**Tracking system and network position server**

The Martin Lighting Director (MLD), a commercial performer tracking system developed by Acoustic Positioning Research and marketed by Martin Professional was used to track the position of performers. General surveys of position tracking systems can be found in [1] and [14]. We developed a network position server in Java that connects to the serial port of the MLD computer to receive continuous updates of performer coordinates and instantaneous velocity. That information is then served to clients either in Cartesian or spherical coordinates with a simple publish/subscribe method using UDP over ethernet for transport. A common task performed by many clients, including the lighting and sound control systems, was the calculation of the distance between a performer and a few common stage points, as well as between pairs of performers. These functions were implemented in the position server itself, so these often used values are only calculated once throughout the entire network.

**Region mapping server**

In many situations, the absolute or relative coordinates of the performer may be unimportant. Instead, the designers or director might simply want to use the actor’s presence in (or absence from) a particular region onstage. For example, in a particular scene, one might intuitively define the regions “upstage left,” “centerstage,” and “near the sofa.” A simple way to visualize these areas would be to shade them on a graphic of the groundplan. That sketch might then be translated by hand to a region that could be expressed numerically in a controller. To create an interactive relationship, a lighting or sound event would be linked to a performer's presence in the region, e.g., “if Banco moves ‘near the sofa,’ raise the lights on it by ten percent.” Why not attempt to eliminate the intermediate step of hand translation to numeric form, and use a sketch on a computer to actually define such a region?

Indeed, the MLD software itself has some facility for drawing shapes on a groundplan that trigger events or adjust lighting output when the performer moves into them. [15] A major limitation with region-based control in this case is the “hard” edge of each region. Actor movement and sensor jitter can cause coordinates to bounce in and out of the region if the tracker is on the boundary, repeatedly triggering responses from the production control systems. This can be avoided by using timers, counters, or low-pass filtering, but specifying an algorithm to set and reset timers or counters becomes tedious. Different low-pass filters might be needed for each situation. Another option is to allow regions with “soft” edges.

---

2 Alberta, Canada: http://www.positioning-research.com/
3 Denmark: http://www.martin.dk/
4 Though it would seem that the first two of these conditions are handled adequately by expressing a relationship in terms of relative position data, it is crucial to remember that the very simple stage directions may mean quite different things to directors and designers in different scenes.
5 The x-y plane of the MLD coordinate system.
6 The MLD can provide “soft” edges but not directly from the drawing tool for the regions. It actually has an extensive set of features for drawing mappings between position data and lighting output. The region mapping server described here is an effort to provide a network-accessible tool that integrates “softness” into the region creation step itself.
In this case, a performer might be in the regions “center” and “downstage,” but if she moves a little to the left, she would still be “center” but less so, and would remain the same amount “downstage.”

The region mapping tool provided a method of quickly creating soft-edged regions and using them for production control. The software, implemented as a multithreaded Win32 application in Microsoft Visual C++, subscribes to position data from the network position server and maps the tracking coordinate system onto a user-defined image containing sketches of each region. Rather than trying to design a graphical interface for drawing the regions, the tool was designed to load an Adobe Photoshop image, parsing and storing its individual layers in memory. As it arrives, each position sample for every performer is mapped into the image, and the opacity of the corresponding image pixel in every layer is made available over the network to other clients. This use of transparency, the “alpha channel” in the image, can be exploited to easily draw soft edges in Photoshop.

These “presence” values returned by the tool can be used by clients in any manner. For example, the lighting system described later might be configured to dynamically map high opacity values to brightness, smoothly fading up lighting intensity as a performer moves through a gradient of decreasing transparency. Different mappings can be loaded on-the-fly by switching the Photoshop image with a network command.

**Interpreted position server**

Additionally, an “interpreted position server” maintained several short-term snapshots of each tracker’s movement and used them to calculate statistics providing some insight into how the performer or prop was moving. These statistics ranged from the familiar—average speed, distance traveled, average direction—to more interesting quantities like the “roundness,” “directness,” and “stillness” of movement. These were continuously calculated for each performer and then made available for subscription on the network. The server subscribes to position data for every tracker, stores incoming samples in a circular buffer, and generates statistics for each performer for a number of different empirically-determined window lengths. Many of these simple statistics perform surprisingly well for control of dynamic elements, especially when a combination of long and short windows are available. The software was also written in Visual C++ as a multithreaded Win32 application.

**Dynamic sound and lighting control**

We wanted to allow experimentation with control over specific details of the cues themselves—lighting and sound intensity, transition speed, choice of media, effect parameters, etc. This required the ability to execute algorithms relating sensor data to the outputs of the production systems. A balance between performance, runtime configurability, and ease of use was desired. The pragmatic goal of having practical lighting and sound control ready for the production of *Macbett* was also very important and led to two different approaches for lighting and sound. A “dynamic lighting system” (DLS) was developed in C++ to allow direct algorithmic control of up to 512 independent lighting channels through a simple scripting language. This approach provided the maximum amount of flexibility but required a considerable development effort and resulted in a fairly unrefined user interface.

Cues in the DLS, rather than defining static looks, define relationships between sensor data and lighting control parameters using user-scripted expressions. The most basic “intensity” expression defines a continuously evaluated intensity for any number of lighting instruments. This expression can be dependent on any combination of performance data variables available from the servers described previously. The dynamic lighting system was designed to be compatible with any traditional lighting console using the MIDI Show Control and Dimmer MultipleX (DMX) standards. An important requirement was that this traditional console could be used during rehearsal and performances as it would
normally. Its control over the house dimmer system could remain intact for any cues that did not need to be influenced by realtime performance data.

Sound control was implemented in the Max/MSP realtime audio signal processing package for the Macintosh. The interface to the different data services available on the network was achieved through a small C-language library, or “external,” that provides rapid parsing of the performance data arriving from the servers. Once the data from the sensing, interpretation, and mapping systems arrived in the Max environment, it could be used to control sound playback, volume, reverb, and any other available processing tools. Some of the most striking possibilities arise from the multichannel sound playback capabilities of Max, which can address up to 512 discrete outputs that can each be connected to one or more speakers, allowing sound to be “placed” anywhere in the theatrical environment based on an algorithm using performer position, speed, or other variables as input.

Production

The UCLA Department of Theater's subscription-series production of Eugene Ionesco's *Macbett* in the Spring of 2001 was the first artistic piece to use these systems in performance. *Macbett* was produced in a largely traditional manner, following the production process typical of large shows at UCLA. It was directed and designed by graduate students, constructed and managed by department staff, with undergraduate students forming the cast and crew.

For the production, a total of five performers were equipped with position trackers; another two were placed in staffs carried by the witches. A network of four workstations provided performance data services for the show. One managed the MLD itself, while the others ran the software described in previous sections: the network position server, interpreted position server, region mapping server, and various support servers. Two additional machines were used for sound control: one provided “traditional” playback, while the other handled “interactive” cues. A final computer ran the dynamic lighting system. (See the block diagram in Figure 1.)

Example interactive sequences

Early in the conceptual development of the show, the director decided that certain performers would be given control over the theater space in scenes where, according to the script, their characters were in some way “supernatural.” For example, Ionesco's two witches influence the environment whenever they appear. In most cases, this influence was tied to a character and if that character appeared more than once, he or she maintained similar control over the environment in each appearance.

A classic insertion of the absurd by Ionesco, the Butterfly Hunter, was to move through a pastoral sound environment that became progressively more detailed as he chased his invisible quarry onto the stage. To accomplish this effect, sound designer David Beaudry used the region mapping tool to draw smooth position-based fades of overlapping audio tracks. Different layers in the image controlled lighting effects in the same scene, fading in new lighting instruments as the actor moved into certain positions.

When the character Macol appears in the final act, a single dynamic cue lights his initial dialogue from a platform at one edge of the stage through his cross to the center to kill Macbett. The stage manager starts the cue when Macol appears on the platform and the operator's go on the traditional lighting console simultaneously triggers both the console's own cue and the dynamic system. The console establishes the scene and platform light for his initial dialogue. When he finally leaves the platform, the dynamic system takes over, opening a path of light before him with each step, while capturing and dimming the rest of the console’s cue as he moves inward. By the time Macol is in the center, he is alone with Macbett in downlight brought in based on his position. A set of simultaneously active dynamic cues handle this sequence based on Macol’s distance from centerstage, matching his timing every night. After Macbett is
killed, the operator takes the next light cue on the Obsession, fading out dynamic control and bringing in the next cue.
parameters of certain cues—eliminates a layer of mediation between action and design. In cases where the “risk” of allowing performers to affect cues is acceptable, the designer can establish relationships directly rather than approximating them with pretimed crossfades. This has fascinating implications for theater and location-based entertainment. So far, these cues have been based on spatial relationships; new sensing technologies will sometime unobtrusively follow human gesture and speech as easily as we tracked position and movement. These will provide information on how performers move and speak, not just where they are.

Using the interpreted position server, this potential could be explored even in Macbett. The clearest example is found in the primary agents of the supernatural, the two witches, who also appear as Lady Macbett and her Lady-in-Waiting. Each witch was to have her own type of control over the environment through her staff. The first conjured thunder and lightning by raising the staff quickly in the air—the quicker and stronger the thrust, the more powerful the lightning strike—while the second witch swirled her staff to create ripples of darkness, color shifts, and the sound of whirling wind proportional to the speed of her staff.

In addition to being based on how the props moved and not where they were, these cues had to remain dormant when the witches moved normally with their staffs. A tracker was embedded in the foam head created for each of the staffs (see Figure 2) and a combination of interpreted position information for both the staff and the witch herself was necessary to achieve the effects. Lightning was created by controlling strobe and intensity attributes of automated lighting instruments while triggering a proportionally amplified sound of thunder in Max. Color shifts combined with slightly randomized intensity control achieved the lighting effect for the second witch. At the same time, the whirling of her staff also faded in and panned the sound of the wind. These relationships were activated at the beginning of each scene where the witches appeared, allowing the actresses to conjure them up at any point.

These cues required the performers to be aware of their new capabilities on stage and to work with the director to explore how they could be most effectively used. An unexpected embellishment (a bug that we made no attempt to eliminate) was the tiny wisp of wind that accompanied certain small gestures of the staff by the actress portraying the second witch. As she learned to punctuate her speech with these gestures in a way that could not be done with traditional cues, we caught a glimpse of the real potential in all of this technology.

In the future, better authoring interfaces will enable rapid creation of basic dynamic cues relating action to complex design elements. However, the possibilities that are most aesthetically intriguing to directors and designers—like the wisps of wind accompanying the witch—will likely require just as much work as in today’s traditional productions. Though the cues were designed and tested before rehearsal, the actors required time in the performance space with active influence over lighting and sound to discover how they could work within the newly responsive environment.

Conclusion

As the novelty of seeing and hearing performers directly affect the stage in Macbett faded and the dress rehearsal process continued, one would occasionally hear in passing, “Oh, we could have done that interactively.” The system performed as hoped: a new creative tool, unpolished but suitable for large-scale performances. The most striking comment came from a well-known lighting designer who saw the performance and took a quick tour of the technology backstage. Despite (or perhaps because of) the relatively basic nature of the performer-controlled cues, one of her first comments was that a completely new genre of performance was possible but unrealized; one that would require a new approach to design. Earlier, a playwright in the cast had suggested that works could be written with these ideas in mind from the start. The director envisioned a workshop process with a few actors, a writer, designers, these systems, and a lot of time. The role of the audience is brought up almost immediately: how can these techniques be
extended to engage them directly in new ways? Our rather traditional performance brings us full-circle back to the performance art and dance experimentation described at the beginning of this paper. The next step is to translate the experience we have gained through this process to new genres enabled by these responsive performance spaces.

This is our motivation for continued systems-level exploration of technology for live performance. From the experience of this traditional production arises the desire to explore what is unique and exciting about the technology’s impact on the process and final result. Engineering research continues to develop devices and systems that have intriguing and evocative artistic applications. There must be real, hands-on collaboration and experimentation with practicing artists in order to understand the challenges and benefits arising from applying such research to live performance.

Figure 2. Witch’s staff with tracker exposed.
Acknowledgements

Professors Fabian Wagmister and John Villasenor provided invaluable advice and assistance. Will Bauer of Acoustic Positioning Research helped us at every step of the way. Thanks to the Macbett production team, especially director Adam Shive, David Beaudry, David Miller, Maiko Nezu, Michelle Magaldi, Katie Mitchell, Ben Story, and Gavin Haag, and the theater staff, especially Jeff Wachtel, Jonathan Burke, and Jeff Segal. Undergraduate students involved in software development were Stephen Lin, John Cheung, A. J. Ross, and Genesan Kim. Partial funding was from the University of California Digital Media Innovations program.

References