Preparation for Improvised Performance in Collaboration with a Khyal Singer

David Wessel, Matthew Wright, and Shafqat Ali Khan (matt.wessel@cnmat.berkeley.edu)
Center for New Music and Audio Technologies, 1750 Arch Street, Berkeley, CA 94709, USA

Abstract
We describe the preparation and realization of a real-time interactive improvised performance carried out by two computer-based musicians and a classical Khyal singer. A number of technical and musical problems were confronted, including the problem of cultural distance between the musical genres, specification and control of pitch material, real-time additive sound synthesis, expressive control, rhythmic organization, timbral control, and the ability to perform for a sustained period of time while maintaining an engaging dialog among the performers.

1. Introduction
This work began as a scholarly study of Indo-Pakistani vocal music (also known as “North Indian” or “Hindustani” classical music) but it quickly became reoriented towards performance. The affinities among us were there and so we began playing together privately in the spring of 1996. After some hours of interacting musically we decided to make some public performances. The first, a duo with David Wessel and Shafqat Ali Khan, took place at IRCAM during a lecture concert in November of 1996. The second, with the full trio of authors came about a week later at CNMAT. The results prompted us to continue. We planned and gave two more concerts in April of 1998 and what follows is an account of the preparations, the technology, and the aesthetic concerns surrounding these evening-long improvised performances.

2. The Musical Context
Musical meetings that combine very distant cultural influences very often end up as aesthetic disasters. Our particular ingredients consisted of a voice strongly grounded in the highly developed Khyal vocal tradition (Wade 1985; Jairazbhoy 1995) and a collection of computer music practices that had little, if any, grounding in a strong musical tradition. Therefore our strategy was to adapt the computer’s role in the direction of the more highly developed Khyal tradition but not completely. After all, two of us had only a very minimal knowledge of and experience with the North Indian and Pakastani traditions and those two of us had no interest in pretending to be well situated in this profoundly deep music culture.

We strove to create a common meeting ground, a situation that would provoke a musical exchange. We did not use another genre with which we had familiarity, such as jazz or rock or 20th century western art music, to make our collaboration some sort of fusion of Indo-Pakistani classical music with another style.

At the same time our aim was not to mimic Indo-Pakistani classical music with modern technology. It goes without saying that we two computer musicians could not give a concert of this music, but even Shafqat acknowledged that he was not singing strict classical music. Instead, we met simply as improvisers, creating a musical space in the moment out of whatever musical abilities and experiences (and technologies) each of us brought to the group.

To allow Shafqat to be comfortable and sing at his best, we took some of the aspects of North Indian classical music as points of reference and departure for our computer accompaniment. Rather than bringing in ideas such as chord changes, modulation, or atonality, we used the drone and the rag as the basis for pitch organization.

We will not attempt to explain or even define the complex and richly developed concept of rag in this paper, but we will attempt briefly to characterize rag in terms of how it structures the use of pitch. We find it helpful to imagine a continuum with “musical scale” or “mode” on one end and “melody” on the other. In both cases there is a particular set of pitches, but in the melody the sequence of pitches and durations is fixed, while in the musical scale no structure is specified for connecting the notes together. Rag would fit somewhere in the middle of this continuum. Rag is more general than a specific melody, because musicians improvise within a rag by singing or playing new melodies. Rag is more specific than a scale, however. Each particular rag has its own rules, character, and history, including different sequences of pitches to be used when ascending or descending, vadi and samvadi, the most important and second most important
notes (which may not be the drone note), characteristic ways of approaching a certain note, famous compositions in the *rag*, and, of course, a collection of pitches.

Our use of rhythm was based upon the Indo-Pakistani concept of *tal*; this work is presented in detail elsewhere in these proceedings [Wright, 1998 #36].

Our performances were to be live, improvisatory, and perhaps the most difficult of all, under control.

3. The Voice

We first made an extensive set of recordings of Shafqat’s voice. We wanted the voice recorded in a very dry manner without a drone and other accompaniment, so during the recording sessions we provided the accompaniment and reverb through sealed headphones. The drone we used was a somewhat simplified version of the one we describe in the next section. We also built a simplified *tal* engine using the CNMAT rhythm engine [Iyer, Bilmes et al. 1997] [Wright, 1998 #36] with a user interface that permitted Shafqat to setup what he judged to be appropriate *tin tal* (16 beat) patterns. For reference we recorded the rhythmic material on a separate track. The result was an isolated and dry monophonic recording of the voice ready for analysis.

For purposes of this paper we will build all of our examples up around *Gunkali*, a *rag* consisting of the pitches C, D-flat, F, G and A-flat. The pitch trajectory shown in Figure 1 is of Shafqat singing a typical phrase from this *rag*. As can be seen the pitch trajectory hits the notes but spends considerable time gliding about. (Sound example #1 is the phrase from which the F0 plot was obtained. It can be heard by clicking on the plot.)

![F0 as a Function of Time](image)

Figure 1. F0 as a function of time. Care should be taken in the interpretation. When the amplitude is very low the estimates are unreliable.

We will return to the pitch profile in a later section. It is at the core of the procedures used for generating pitch material in the accompaniment. We also analyzed these recordings to obtain data sets for our additive synthesis system.

To get a better idea of the precise pitch content of Shafqat’s improvising in *Gunkali*, we produced a histogram of the amount of time spent on each pitch. This histogram is shown in Figure 2 and was collected over several seconds. The use of the time-on-pitch histogram was motivated by the work of Krumhansl (Krumhansl 1990) on the cognitive representation of musical pitch. One of the striking features of Krumhansl’s histogram or pitch profile approach is that it portrays some of the most perceptually salient features of a pitch system and has been shown to be useful for the characterization of pitch organization in North Indian classical music [Castellano, Bilanuci et al. 1984]. Krumhansl’s plots were all generated with pitch classes along the horizontal axis. Given the extensive use of pitch glides in our vocal samples, a much finer frequency resolution was required. We choose to place the histogram intervals at 1 Hertz intervals and
as can be seen in Figure 2 we recovered the notes of *rag Gunkali*. An interesting feature of this pitch profile is its accurately tuned character. Even though the pitches glide about considerably the peaks are very sharply tuned. We would not see such sharply tuned pitch peaks from a western vocalist using periodic vibrato.

![Pitch Profile Diagram](image)

Figure 2. A time on pitch histogram for a 22-second segment of *rag Gunkali*. The five highest peaks correspond to the pitches (C, D flat, F, G, and A flat). The histogram bin size is 1 Hertz.

4. The Performance Situation

![Concert Setup](image)

Figure 3. A photo taken during a sound check for an April 1998 concert gives an idea of the stage setup. The concert took place in CNMAT’s intimate performance space with a maximum capacity of about 60 persons. This space is equipped with an 8 channel sound system based on John Meyer’s HM1 speaker technology and an additional 2 channel system using Meyer’s UPL-1 speakers. We configured the 10 channels of the sound diffusion system for a frontal sound and a surround reverberant field. We assigned 4 speakers to the direct sound and the remaining 6 to reverb. In the layout of the frontal sound we sought a wide centered image for the singer and offset stereo images for each of the computer performers. We sought a natural acoustic quality.

We placed all computers out of the performance space to achieve a minimum in distracting noise and visual clutter, except for a silent Macintosh Powerbook used simply to provide a visual display of the current state of the rhythmic software.
With the exception of rhythmic synchronization, the two computer musicians performed independently of each other. David Wessel’s setup included two controllers, a 16 channel MIDI fader box and a Buchla Thunder providing for poly-point continuous pressure and location control and variety of selection mechanisms. A Macintosh running the MAX programming environment was placed in between the controllers and the EIV sampler. Rhythmic synchronization was achieved by slaving Wright’s rhythm engine to Wessel’s tempo.

The setup for Matt Wright was a bit more complex. His main controllers were a Wacom tablet (Wright, Wessel et al. 1997) and a 16 channel MIDI fader box again linked to a Macintosh running MAX and equipped with SampleCell cards. The tablet has a clear plastic overlay under which we placed “template” that visually depicted each of the regions of the tablet’s surface for which we defined behaviors. We are thankful to Sami Khoury for creating software to help lay out and print these templates. Max used the OpenSound Control Protocol [Wright and Freed 1997] via Ethernet to control CNMAT’s additive synthesizer CAST running on a pair of SGI computers. We used CNMAT’s building-wide audio patching system to bring the sound from the SGIs in the basement machine room up to the mixer in the performance space.

Shafqat’s voice was amplified and treated with reverb.

5. The Drone

An important feature of Hindustani classical music is the constant drone provided as a tonal reference. In traditional acoustic settings, this drone is usually provided by a stringed instrument called the tamboura, which is played simply by plucking each of the 4 or 5 open strings slowly in sequence, pausing, and restarting. Our synthetic drone instrument began with a pair of four-second sound file excerpts of groups of tambouras droning. We analyzed the excerpts with CAST analysis software to produce additive synthesis datasets.

The first incarnation of the synthetic drone was for a concert on 11/15/96 that was supposed to be a duet between David Wessel and Shafqat Ali Khan. Hours before the concert we decided to add a drone aspect to the piece and control it from the Wacom tablet. For this instrument, the idea was to simulate the gestures used by tamboura players. We defined 6 virtual strings as regions on the tablet surface, each of which corresponded to an additive synthesis voice resynthesizing one of the tamboura data sets. A “pluck” gesture caused the corresponding voice to play the data set. We wrote software to analyze the shapes of these pluck gestures, for example, starting and ending vertical position within the region and the kind of motion made with the pen during the gesture. We mapped these gestural parameters to synthesis parameters controlling timbre, for example, the balance of even and odd harmonics.

For later concerts, we designed a "drone auto-pilot" that would automatically manage the repetitive aspect of plucking the virtual strings in turn. We wanted to retain the timbral controls that were so effective in the earlier instrument, so we moved to a model where each timbral parameter has a global value that can be adjusted in real-time, and each automatic pluck takes the global value of each timbral parameter. To avoid monotony and provide for continually unfolding richness without manual control, we added a small random jitter to the timing between plucks and to the values of the timbral parameters for each pluck. Sound example #2 illustrates the basic drone.

Another refinement to our drone instrument was the addition of sinusoids one and two octaves below the fundamental. Originally, these were synthesized as constant-frequency sinusoids with manual control of amplitude. This proved to have an undesirable effect, adding a “synthetic” sounding static quality. Another effect of these static sinusoids was quite amusing in retrospect: because the frequency of the one-octave-down sinusoid was nearly 60 Hertz, the sound engineer thought there was a ground loop. We solved these problems by using the amplitude and frequency trajectories from the lowest partial of one of the analyzed tamboura samples, transposed down. This added detail and "life" to the low sinusoids; sound example #3 illustrates the drone with added low components.

As a final twist, we added some of the character of Shafqat’s voice to the drone instrument. We analyzed an excerpt of him singing the drone note and used CAST’s timbral interpolation mechanism (http://cnmat.CNMAT.Berkeley.EDU/CAST/Server/timbralprotos.html) to interpolate the timbre of his voice with that of the tamboura on two of the virtual strings. Sound example #4 illustrates this "voice-morphed" drone.

6. Rhythm, Pitch, and Timbre for the Poly-Point Interface

David Wessel’s software was designed to control rhythm, pitch, and timbre with a poly-point continuous controller for which he used Buchla’s Thunder. Eight distinct algorithmic processes ran in parallel throughout the performance. Each of the eight processes was associated with a pressure-by-location strip on the controller.
Applying finger pressure to the strip brought the underlying process to the sonic surface and changing the location of the finger along the strip performed a timbral interpolation. Additional surfaces on the controller made it possible to select among a variety of rhythmic and timbral structures. As these rhythmic structures were know to the performer, he was able to select out individual notes and groups of notes by applying pressure at the appropriate times. We have come to call this *dipping* as the algorithm remains silent unless the pressure gesture is applied. Unless the performer is actively engaged with the controller all sound stops. Slow crescendos and decrescendos are easy to execute, as well as rapid entrances and departures. Notes, fragments, and whole phrases are selected from an underlying time stream and precise timing is maintained by the underlying rhythmic processes.

The eight algorithmic processes were distributed across different registers. As the control strips for each of the processes was located right under the fingertips of both hands, the performer could easily manage the registral balance. (This aspect is used extensively in the performance excerpt sound example.)

In the underlying algorithms pitch profiles controlled the probability that a given pitch would occur and were designed to accommodate the frequency profiles as shown in Figure 2. While pitch profiles were applied to the pitch classes, rhythmic profiles were applied to the *tatums* of the underlying rhythmic cells [Iyer et al. 97]. The shapes of the profiles were controlled by selection operations and by a non-linear compression and expansion technique. Location strips available to the thumbs allowed for the control of the shapes of both the rhythmic and pitch profiles. When profiles were expanded the differences among the values associated with the pitch and rhythmic probability arrays were exaggerated and when they were compressed the profiles were flattened. This proved to be promising way to control density in the rhythmic structure while maintaining its structural integrity. It also facilitated a control of a widened or focused pitch palette.

Other rhythmic features had profiles associated with them. Most notable were the deviation arrays associated with each rhythmic cell. Here temporal deviations from isochrony, as in the long-short temporal patterns of swing, could be compressed, that is, flattened towards isochrony, or exaggerated. Another important profile controlled the actual durations of the notes, not the time between the onsets of the notes. Operations on this feature allow the performer to move from a *staccato* type phrasing to a more *tenuto* one in a smooth and expressive manner.

We have developed a strategy for representing hierarchically structured data in MAX in spite of its paucity of data types, using the *refer* message to *coll*. MAX's collection object. Therefer message causes a *coll* to replace its contents with those of the *coll* named in the argument to the *refer* message. The *coll* object stores a "flat" set of data, which we use to represent different orchestrations, rhythmic patterns, and other behaviors of the algorithms. By storing the names of our underlying *coll* objects as data in another collection, we can treat entire collections of data as atomic references, much in the way programmers in other languages can store and manipulate a single pointer that refers to an arbitrary amount of data. We use another *coll* as a sort of buffer, sending it *refer* messages from our master collection, which allows us to switch among complex behaviors with a single message. The referencing of collections of data in MAX is implemented with pointers, so it is efficient and provides reactive performance even when massive changes in the data used by an algorithm are engaged.

7. Scrubbing Through Additive Synthesis Data Sets
Matt Wright accessed additive synthesis data sets with a Wacom tablet interface. We analyzed a series of sung phrases from the recording sessions with the CAST tools. The time axis of each of these data sets was laid out on the tablet surface so that the Wacom pen could be used as a scrubbing device. The high-resolution absolute pen position sensed by the tablet was mapped to a time in the data set so that at each instant the data being synthesized was determined by the pen position. Moving the pen steadily from left to right across the tablet such that the time taken to traverse the entire scrub region is exactly the length of the original phrase resynthesizes the original material at the original rate. Moving the pen at other rates, or backwards, naturally plays back the phrase at a different rate. When holding the pen at a fixed point, the synthesized data becomes a very synthetic sounding static spectrum taken from a single instant of the original phrase. When there is pitch deviation in portion of the analyzed phrase corresponding to the area immediately around the current pen position, a slight vibration of the pen position causes a vibrato. We found that even a tiny wiggle of the pen was enough to induce enough variation to avoid the problem of the static spectrum.

Bringing the pen to touch the tablet in the middle of the time axis started the resynthesis at the given point, and taking the pen away stopped the sound. We added some envelopes to fade the sound in and out gradually in these situations so that the entrances and releases made by the pen would have a natural quality.
In the interface used in the first concert, there was a single large area for this scrubbing operation, and a palette of data sets that could be selected. We found this quite difficult to control, because it required perfect memory of the contents of the analyzed phrases in order to find the desired bits to play or even to play in tune. For the second concert, we moved to a model where each data set to be scrubbed had its own region on the tablet. The width of these regions still took up almost the entire tablet, to maintain high resolution control of the time axis, but their height became compressed as much as possible. With a fixed region of the tablet surface for each data set, it became possible to draw some of the features of each phrase on the surface of the tablet. We marked regions where one of the notes of the *rag* was sustained, drew curves to represent pitch contours, and wrote the syllables of the sung words.

7.1 A Tracking Filter-like Effect

The Wacom interface was also used to control the spectral content. We have found it very effective to bring to the forefront a particular harmonic of a vocal line. The expressive character of the pitch and amplitude contour is maintained but a whistle-like effect is produced. Because of the importance of playing only those pitches compatible with the *rag*, we selected only the harmonics whose frequencies were octaves of the fundamental. This technique was implemented in the additive synthesizer in a manner analogous to a parametric equalizer except that the spectral shape tracked the fundamental frequency. The pen pressure sensed by the Wacom tablet was used to control this feature. Sound example #5 demonstrates this scrubbing technique with continuous control of the tracking filter-like effect.

8. Rhythmic Control from the Tablet

Our approach to rhythmic control from the tablet took advantage of the strengths of the tablet and complemented the control afforded by Thunder. Whereas the emphasis of the Thunder interface was on real-time control of precomposed material, the tablet’s lack of poly-point control made this kind of “orchestra at the fingertips” interface impossible. Instead, we took advantage of the tablet’s high-resolution absolute position sensing and our templates to define hundreds of small regions on the tablet surface; these allowed us to construct arbitrary new rhythms to be played on the next rhythmic cycle.

The centerpiece of the tablet’s rhythmic control was a grid of sixteen boxes arranged horizontally, corresponding to the sixteen beats of the rhythmic cycle used as our basic framework. We used a “drag and drop” interface to select a preprogrammed rhythmic subsequence from the palette and place it onto one of the beats of the rhythmic cycle. The individual regions of our palette were large enough for us to draw rhythmic notation on the template, allowing us to see what subsequence we were selecting.

We controlled the selection of particular percussive timbres from a separate section of the interface. The subsequences were defined in terms of abstract drum timbres. Part of the tablet surface was a palette of the various collections of samples used for percussion synthesis; these were associated with the abstract drum timbres via another drag-and-drop-style interface.

The environment for rhythmic control is described in more detail in a separate paper in these proceedings [Wright and Wessel 1998].

9. Conclusions

We provide a final sound example (number 6) which demonstrates the results. Each concert was a full evening consisting of four works each based on a different *rag*-derived pitch collection. We have plans for another round of concerts in the fall of 1998 and it would seem appropriate to make a brief assessment of the work so far and what we plan to alter and add in the future.

The most important observation is that when one designs instruments that can be played with a reasonable degree of control intimacy, lots of practice at performing becomes essential to a musical result. This implies that software development impacting the control interfaces must cease long in advance of the actual performance. We have found it particularly difficult to balance the time spent in software development and that spent playing.

We would like to have more flexibility along the continuum between “scale” and “melody.” Resynthesis of prerecorded material gives wonderful flexibility in altering the timing and timbre, but we are pursuing techniques for generating less constrained musical material in a continuous manner [Wessel, 1998 #35]. Another feature that we plan to develop further concerns the representation and control of pitch glides or bends, one of the key features of the genre.
It is humbling to share the stage with a master musician such as Shafqat Ali Khan. In the improvisatory context, musically mutable material must be available at all times. The facility with which a trained singer can draw from a repertoire of known material in each moment of a performance makes our attempts to organize and access musical material by computer seem clumsy and frustratingly slow. A singer’s ability to react almost instantly to what is heard or imagined defines a standard for low-latency reactivity that is still well beyond our current capabilities with computers. A large repository of material is essential as well as reactive devices for exploiting it. Unfortunately, the common practice of preparing a piece for the traditional linear exposition of a work is of little assistance here. Our results to date inspire us to continue to improve our tools for using computers in improvised performance.

References


List of Sound Examples
[1] A typical phrase from Rag Gunkali, as sung by Shafqat in a dry, isolated recording. (5 sec)
[2] The basic additive synthesis drone, taken from tamboura samples. (20 sec)
[3] The drone augmented by extra sinusoids one and two octaves below the fundamental of the original samples. (30 sec)
[4] The drone augmented by timbral interpolation between the tamboura and Shafqat singing the drone note. (21 sec)
[5] Short performance excerpt demonstrating scrubbing and control of the whistle-like effect from the tablet. (17 sec)
[6] Longer performance excerpt. (76 sec)