"INSTRUCTIONAL APPLICATIONS OF COMPUTER MUSIC"

Howard A. Puelles
School of Education
University of Massachusetts
Amherst, MA 01003

and

Stuart Smith
College of Music
University of Lowell
Lowell, MA 01854

ABSTRACT

This paper is intended to stimulate thinking about instructional applications of computers in music. To date, computer music research and development has concentrated on technologies primarily for specialists -- composers, theorists, musicologists, etc. Believing that there is a broader market waiting, we assert the importance of taking an educational perspective in the design of computer music systems -- systems that will meet the needs of a wide range of musicians, including amateurs and professionals, students and their teachers. We identify some desirable goals for such an educational computer music system: powerful compositional facilities, music data processing functions, on-line music libraries, better human factors engineering, etc. To illustrate, we describe a recently developed prototype music instruction system which was implemented on a large time-sharing system. Further, we speculate about ideal design features for a self-contained, hand-held microcomputer-based music system. We conclude by listing some educational issues to keep in mind as computer music systems continue to evolve.
"INSTRUCTIONAL APPLICATIONS OF COMPUTER MUSIC"

Robert A. Peelle and Stuart Smith

INTRODUCTION

Computer-based instruction in music has been growing embryonically during the last decade. Computer music systems are in use at a few colleges, where students' performance in basic musical skills is reported to be consistently higher than that of students receiving only conventional instruction. [1] It seems obvious that there is great potential for instructional applications of computers in music.

Yet the computer has had only marginal impact on the teaching of music so far. Why? To begin with, computer music systems are generally too expensive, too difficult to use, and controversial artistically. Furthermore, educators tend to resist technological innovations—particularly those purported to be complex and threatening—like computers. Indeed, there is a noticeable lack of computer literacy among the general public, and music teachers are certainly no exceptions. Instructional applications of computer music could be significantly more developed than at present, and it is time for educators to become involved in the design of computer music systems.

GOALS FOR AN EDUCATIONAL COMPUTER MUSIC SYSTEM

From an education perspective, it is desirable for a computer music system to be:

- portable
- inexpensive
- easy to use
- musically appealing
- capable of teaching
Such a portable system should be lightweight (say, under a kilogram), basically self-contained (no connecting wires), and, preferably, small enough to hold in one hand. It should sell for about $50 or less (1980 currency). It should be easy to get started and to obtain musical results immediately; it should be easy to understand what is going on and how to make things happen; and it should have good human factors design. It should appeal to a wide range of musicians — including both amateurs and professionals, students and their teachers. It should be equipped for teaching and learning in a variety of modes, such as drill-and-practice, tutorial, simulation, gaming, laboratory experimentation, problem-solving, and creative/artistic projects. (See [1] for detailed description of instructional applications of computers.)

More specifically, we would like an educational computer music system with facilities for:
- playing
- listening
- composing
- orchestration
- conducting
- programming
- teaching

This means we need real-time sound (hopefully of higher quality than MERLIN™), for playing pieces or just tinkering around. Plug-in modules can provide on-line libraries for listening to a variety of musical pieces, as well as for lessons, games, and exercises. For composition, orchestration, and conducting purposes, specific functions are needed to create, store, play, and generally manipulate musical entities (at a number of levels). And, of course, we want a high degree of user control; one should be able to write programs in a music-oriented language — not a general-purpose programming language.* Furthermore, the user should

* With all due respect for LOGO. (See [3] in this volume.)
also be able to write teaching programs, using the computer as an aid —
that is, the system should allow anyone to be an author.**

DESIGN OF A MICROCOMPUTER-BASED INSTRUCTIONAL SYSTEM

In this section, we describe a microcomputer-based instructional music
system designed to meet some of the goals mentioned above. This hypothetical
device is inspired by the ubiquitous hand-held calculator.

Technology now exists which makes such a pocket-sized computer music
system realistic. Earlier this year, Harold Alles reported a design for a
12-oscillator digital synthesizer which could be realized as a single micro-
processor-sized integrated circuit chip with current production methods. [5]
Also this year, Hewlett-Packard announced custom versions of their HP-41C
calculator: if a customer's application requires a large number of calculators,
H-P will implement a special function set in read-only memory and fabricate an
accompanying labelled front panel overlay. [6] (Incidentally, the H-P 41C
contains an internal tone generator that can be programmed to play simple tunes.)
Also, Radio Shack recently announced a pocket-sized computer with a fairly
concrete implementation of BASIC. [7] The cost of calculators is now low enough
so just about anyone who wants one can own one. Most calculators are relatively
easy to use; even some elementary school children —
use them wherever and whenever they feel like it. And now, calculator-style
sound-producing electronic devices, such as Texas Instruments’ Speak & Spell,[8]
are available on the market for learning (in and out of school).

The music calculator we envision is pictured in fig. 1. In brief, the user
of this device can play and/or record notes by keying them in on a miniature
keyboard at the bottom, with sound output provided by an internal synthesizer
like Alles’ proposed one-chip device. Playback of recorded music is controlled

** Nygride’s OUTDO system has powerful music-creating features,
but, because of the underlying PLATO protocol, students do not have access
to any of the authoring facilities. [4]
by the seven circular buttons -- PLAY, STOP, FAST FORWARD, REWIND, and RECORD -- as found on virtually all tape cassette recorders, plus note-by-note single-stepping functions commonly found in electronic music "sequencers". Potentiometer knobs (on the side) provide real-time control of tempo and volume.

The proposed music calculator has several kinds of features which we feel are particularly important for computer music systems intended for use in teaching. These are:

On-line libraries of musical composition: The music calculator has provision for plug-in (KM) modules containing a variety of interesting and useful pieces. These pieces are accessed through a "hum-a-tune-halt" scheme: the user keys in the first few notes of the desired piece, then the system finds the piece in the library and brings it into working memory. (The library search algorithm attempts to find the best match between the user's input and pieces in the library; if a result, having errors or imperfect recollection of the desired piece is not necessarily present, the piece from being found.)

Transformation functions: The user can transform the piece in working memory by applying any of several available functions; among the functions provided are: inversion, retrograde, augment/diminish, transpose up/down, horizontal/vertical flip, shuffle (randomize), and contrapuntal combination. There is also an "undo" function in case the user uninintentionally makes a piece and wishes to restore it to its condition just prior to the last transformation.

Two levels of programmability: Users can create, save, and run programs consisting of any combination of the functions and data elements represented by keys on the front panel. In addition, there is a second level of "privileged" functions available to professional programmers who wish to create computer programs. Unlike the user-level functions, which perform primarily musical operations, the second level of functions perform programming tasks such as setting a flag, decrementing a counter, testing for a condition, matching, branching, etc. This second category of functions is necessary for programming lessons, games, and exercises which may be embodied in plug-in modules.

A prototype instructional computer music system

In order to test out these ideas, we developed a prototype instructional computer music system consisting of a special-purpose keyboard, a software simulation of the music calculator functions, and a single four-voice tone generator. The software was written in ML and ran on a CYBER 117 time-sharing system at the University of Massachusetts. A CRT terminal was used to sign on and sign off the time-sharing system, but during sessions the special-purpose keyboard was used exclusively for interaction. (See [8] for details.)
About forty students at the University of Lowell and the University of Massachusetts tried out the prototype system. This pilot study yielded some suggested uses for an eventual hand-held public calculator. Among these were the following:

Pocket utterance/reader -- a source of accurate pitches and durations.

Composer's notebook -- to hold musical ideas for later playback and use.

Music student's assistant -- to play a tricky rhythmic pattern, an ear-training exercise, a particular musical passage, etc. -- anything one finds difficult to play or sing.

Diction exerciser -- for either preprogrammed or user-programmed exercises.

Automated pocket reference manual and thematic locator -- to discover the identity of a musical theme, to find examples of particular ornaments or cadence formulas, to find compositions that all contain the same melodic motif, etc.

Automatic composer -- to produce a variety of stochastic compositions automatically (in connection with library or user-supplied musical material).

Logic games -- for instance, to pose the problem of transforming a given pitch pattern into a different one using only the transformation functions (i.e. no keyboard or library input).

Much more experimentation with this prototype computer music system is conceivable. For instance, we would like to study further how a wider variety of users -- including children, concert pianists, grandmothers, janitors, etc. -- would use the various features. We would, of course, prefer to conduct such studies without the constraints of a host time-sharing system. We would also be interested in investigating the effects of using a better synthesizer; perhaps the prototype could be linked to a Synclavier™. And perhaps it could be enhanced to do multitasking -- adding additional voices while earlier ones are playing, or manipulating compositions in real time -- to become a "super conducting" system.

67
In this section, we briefly describe features which we think would be desirable in future education-oriented computer music systems. Among these features are:

*Haptic Shape* We envision a device shaped so that it conforms to the human hand, to be held comfortably, it would have an elongated tubular shape, tapered at one end, encased in soft plastic with rounded contours — in contrast to the regular rectangular hard plastic/metallic calculators and electronic games of today. For convenience we, it should have controls configured to correspond to natural physical movements of the wrist and fingers — with particular attention to the thumb-forefinger juxtaposition.

*Control Mechanism* We envision a variety of mechanisms which provide the user with control over musical utilities. For instance:

- **Mnemonic Keys**
  Symbols inscribed on the top of each finge-tip-sized key suggest their musical functions; e.g. F for resonating notes.

- **Finger Sensor**
  Functions and commands would be affinity-grouped and perhaps color-coded; e.g. the commands for PLAY, RECORD, STEP FORWARD, and STEP BACK would be placed together and have similar pictorial symbols.

- **Pressure Sensors**
  Volume could be controlled by squeezing the sides of the device with one hand. Increased pressure would cause louder sound production, and a LOCK button could be pressed with the other hand to hold the volume at a desired level and allow the user to release pressure.

- **Real-time Notation Controls**
  Tempo and pitch could be controlled by rheostat-like dials.

- **Incriment Constructor**
  Switches for creating, selecting, combining, and storing different timbres — plus an oscilloscope-type display panel for viewing the spectra.

- **Keyboard**
  An accordion-like keyboard (along the length of the device) for playing and entering notes — with automatic adjustment to designated temperaments and with facility for referring to different octave registers.

*The device is purposely not pictured here in order to leave room for the reader's imagination.*
- Articulation Control
  Note durations can be adjusted (independently of tempo) to achieve staccato, legato, etc.

- REST Bar
  Rests are entered like notes by using a horizontal key found below the white notes (like the space bar on a typewriter).

- Motion Sensor
  Rhythm can be computed by waving the device like a wand.

- Key Transposer
  By holding the KEY button down and sounding a note, the entire piece is transposed to a new key.

- UNDO
  This button (recessed in the bottom of the device) returns the current musical piece to its previous state (e.g., just before an unintended use of a function). That is, it will undo the effect of the last key press.

- Mixer
  Basic mixing facilities for selecting, combining, and editing different tracks or layers of sound.

- REAR Again
  A handy, thumb-protected button automatically plays back the last measure or so.

- REPEAT Refrain
  In constructing a piece, any designated section could be punched in -- without re-entering the notes. Indeed, these "sub-scores" [9] could be given names (like sub-routines) for convenient repeated use.

- Privileged Functions
  Function keys could serve double duty, with pop-up labels revealing privileged functions for more sophisticated users such as curriculum authors, who may want to use matching and branching facilities for computer-assisted instruction, or professional musicians who may want to do statistical analyses.

- Graphics
  A display screen with a command menu (like Smalltalk [10]) and scrolling, etc. like standard text-editing facilities.
- Peripherals

Optional hook-ups to a full sized keyboard, printer, amplifier, headphones, electronic light generators, one-meter x six tablet, or other computer music synthesizers (for "microsymphonies").

Notation: we believe that now is the time to reconsider music notation.

This time of transition to computer-mediated music is a rare opportunity to recast and improve conventional notation. The increased power and control afforded by computers purports to expand significantly the scope of music (and perhaps change its very nature). Surely this warrants new notation.

We realize that this suggestion threatens to open an old Pandora's box and that previous proposals to change musical notation have met with severe resistance. Nevertheless, we foresee that computer music will soon outstrip the capabilities of conventional notation (which has already shown signs of stress). Accordingly, we suggest embarking on a new design effort which anticipates the notational needs of music in the future.

Essentially, we want a language for expressing "all things bright and beautiful" in music. We want to be able to represent symbolically musical entities ranging from the envelope, pitch, time, and duration of individual notes to melodic phrases and themes, to formal sections, movements, and whole pieces. We also want to be able to refer to abstract, "out of time" musical structures -- scales, chords, tempi, stereos, and the like -- simply and naturally (that is, without necessarily engaging the underlying physics or mathematics). And, we want to be able to note effects such as crescendo and diminuendo, which vary continuously in time. Further, we would like a notation which makes it possible to express higher order musical phenomena such as compositional methods, abstract formal schemes, and even strategies for teaching music.

Ideally, the notation would optimize a number of desirable characteristics, including:
- Simplicity
- Practicality
- Power
- Aesthetics
- Education

By simplicity, we mean concise, unambiguous, and uniform syntax. By practicality, we mean familiar symbols for functions which are useful, flexible, and natural. By powerful, we mean that the notation is immediately executable (by computer) and versatile in its applications. By aesthetic, we mean a graphic notation which is mnemonic, suggestive, and elegant. By educational, we mean easy to learn and helpful in teaching.

While design considerations for such a notation are certainly challenging, we might gain some insights from designs of current programming languages such as ALP (A Programming Language) which was conceived as a unifying mathematical notation and which embodies similar general principles. [11]
The general question remains: what notation can represent the maximum range of musical phenomena and provide an effective tool for both the professional musician and a young child learning music for the first time?

**Artificial Intelligence.** We would like to see a computer music device endowed with some artificial intelligence — particularly these qualities which enhance computer use in instructional applications. For instance:

- **Automatic Error Correction**

  E.g. Command syntax: the system would make a best guess at what the user intended to key in and execute it even though the command syntax was faulty. If the guess is wrong, the user can always halt execution and re-enter the command. (This is similar to the problem of detecting and correcting misspelled words by a machine in a natural language context.)

  E.g. Musical syntax: the system would identify obviously wrong notes — say, a chromatic note in an otherwise wholly diatonic context — and substitute the nearest plausible note; or, the system would correct a note which doesn’t fit with the prevailing chord.

- **Diagnosis**

  The system would identify patterns of learning problems, anticipate when they might occur again, and prompt the user accordingly. It could prescribe remedial exercises, or suggest experiments to try, or, perhaps, correct the errors in real time for the student.

  E.g. In ear training, the system would report that the student is consistently failing to discriminate perfect 5ths from perfect 4ths; or, in performance, that s/he is consistently misplaying (and misplaying) a particular note or group of notes.

- **Self-teaching**

  We would like a system capable of explaining itself (at several different levels). It must understand something about what it is doing — both in performing music and in conducting instructional programs. For instance:

  E.g. Built-in operating manual: the system would contain instructions on how to operate the device, so that anyone could pick it up, try it, and immediately learn more about how it can be used.

---

*These capabilities raise an important pedagogical issue regarding interruption of students’ learning processes — that is, what to interrupt and offer advice, hints, or instruction vs. when to shut up and let the student struggle or discover things on his/her own. The issue becomes most poignant when the instructor corrects student errors without the student’s knowledge. This is a problem in teaching whether it be with a “mass” machine or a human instructor.*

71
E.g. Example tunes: each key could play, say, a 3-second pre-
programmed musical tune illustrating the particular function.
Perhaps these examples could be activated by poking a pin or
pencil or fingernail into a small hole in the center of the key.
E.g. HELP: depressing the red HELP button would call for assistance
while one is learning how to perform a particular skill. The
system might provide a lead-in phrase, play accompaniment, or
fill in parts of the piece.
- Transcribe Live Performance

We would like a system which would allow the user to sing, whistle,
or otherwise play a tune which would be transcribed into computer
representation. (We acknowledge the difficulty of this task and
refer readers to Aby Moore's psychosomatic research with single
melodies. [4]. We also wish to emphasize the pertinent educational
question of how much to normalize the music.)

APPLICATIONS FOR EDUCATION IN THE FUTURE

The development of computer music systems may have profound effects on
education in the future. Consider the following open questions:
- How will the general availability of computer music systems affect
  music education?
- Will they necessarily improve conventional music instruction?
  With such advances, how will students prefer to learn music?
  In the classroom? Out of the classroom? Will their music competence
develop more quickly or less quickly? More thoroughly or less so?
Will it last?
- How will society cope with greatly increased numbers of (non-skilled)
  musicians? Will music become a "weaker" subject?
- How will the roles of teachers change?
  Since the computer can carry out drill-and-practice quite effectively,
  what will teachers do? Will they shift their efforts to guiding,
  motivating, and inspiring students? Will they perceive the computer
  as a competitor or will they seek to establish a partnership --
  perhaps delegating those things that computers do well to computers,
  and those things that humans do well to themselves?
- Will teachers invent new uses of computers in music education?
- How will students' roles change?
  What will students do when they can act like composers, conductors,
  orchestrators, studio producers? Or critics, analysts, or treasurers?
How can one tell who is the building performer and who is the music
listener-junkie?
How will computer music augment communication systems?

References


