CONTOUR: A REAL-TIME MIDI SYSTEM
BASED ON GESTURAL INPUT

Arne Eigenfeldt
Department of Music
University of Victoria
Victoria, B.C. CANADA

Abstract

CONTOUR is a real time improvisatory/interactive musical software system which generates MIDI (Musical Instrument Digital Interface) information based on gestures rather than specific details. Phrases can be created, varied, and combined in real-time with a minimal amount of data specification: the power in the system resides in its ability to create music through continual interpretation of graphic gestures. Gestures in the system control pitch, volume, rhythm, and continual and independent transformations of these functions.

CONTOUR specifically explores two musical features: improvisation and the use of pulse and metric rhythms. Working on the assumption of patterns, repetition, modality, and pulse, a series of independently repeating gestures are created: the details of these patterns are continually varied while maintaining gestural consistency. Through the use of subdivisinal and accent weightings, a level of rhythmic complexity can be created that emulates that of cultures outside of the European tradition. However, these rhythms maintain a naturalness usually not found in computer-based music.

CONTOUR is complete in this version, and has already been used by the author in several compositions for performance and dance.

Overview

CONTOUR is written in MACH 2, a subroutine-threaded FORTH with automatic macro substitution, and runs on an Atari 1040. It generates MIDI data which is sent to external MIDI controlled synthesizers. The initial design of CONTOUR was influenced by portions of HMSL, the Hierarchical Music Specification Language, (Polaneky and Greenboom, 1985). Like portions of HMSL, CONTOUR represents information as a series of histograms and graphs, thus allowing for the creation and manipulation of gestures, or contours, rather than only the raw data they represent. CONTOUR does not allow the user to

101
access specific data; instead, the user is presented with a series of graphic representations.

ConTour creates "instrumental" music in that it attempts to model certain aspects of human performance (the notion of MIDI as note-oriented is accepted with open arms): monophonic patterns are combined to create ensemble type gestures.

ConTour is comprised of four distinct sections: Create, in which contours are created; governing pitch, volume, and rhythm; Transform, in which contours are created to vary the contours in Create; Quantize, in which transformed data from Create is modified to conform to predetermined structures, such as mode and rhythmic tendencies; and finally Perform, in which quantized, transformed data from Create is combined and sent to external synthesizers via MIDI.

Data Creation and Representation

The user creates independent gestures, or contours, which represent pitch, volume, timings, and other functions, either through mouse drawing, or various forms of random procedures. Because ranges and offsets are assigned to individual contours, similar gestures may represent diverse data.

Patterns

Fundamental to ConTour is the use of patterns and repetition: the resulting music is a series of independently repeating gestures. When contours are drawn, loop points are set through the use of moveable virtual sliders, independent for each contour. Because loop points and pattern length are independent between all contours within ConTour, continually shifting combinations of patterns can be achieved.

Each contour may progress at its own rate, depending upon the values set, or they may be linked. In either case, the gestural premise of the system is clearly demonstrated: a basic shape, be it melodic, dynamic, or rhythmic, is maintained throughout, yet with continually varying details.

Modality

Up to 64 modes can created and stored by selecting notes from a software keyboard conforming to the MIDI standard (1983). Values from the pitch contour are compared to this mode and altered accordingly.

Transpositions are possible after quantization. Pitch patterns remain the same in a transposed mode because the quantization is transposed, not the raw data.

Pulse and Metre: Rhythmic Quantization

Pitch gesture and its quantization is a straightforward matter, and the user can easily hear the correlation between a contour and its resulting performance. Unquantized time
gestures are equally as recognizable. Because metric music deals with the concept of beat, certain considerations had to be taken into account; unfortunately, these considerations sometimes blur the correlations between contour and resulting rhythm.

Contour is based on an understanding of pulse rather than subdivision. Rhythmic complexity is derived from individual patterns sharing a common pulse; however, differences in subdivision – such as triplets versus quarters – may easily be defined. Subdivisions of a chosen pulse are created through graphs and assigned weightings representing potential subdivisions. Data is taken from the user created rhythm contour and compared to the pulse and the available subdivisions – subdivisions are chosen first by their proximity to the raw data. Following this, a random number is compared to the percentage weighting of that subdivision: if the weighting is higher, the subdivision remains; if lower, a new subdivision is chosen based on the weightings of all subdivisions. The subdivision remains throughout a rhythmic phrase: this phrase length is a user defined percentage of the rhythmic contour.

The musical result is a constant pulse with continually shifting metric subdivisions. Although predefined rhythms are impossible, clear rhythmic gestures can be created.

Velocity Quantization

Accented phrases within the pulse can be created using volume quantization. In a process identical to that of selecting rhythmic subdivisions and subdivision phrasings, accent phrases can be set. When a phrase begins, an accent pattern and phrase length is chosen. On the first occurrence, the raw data from the volume contour is considered the accent, and the remainder of beats in the accent pattern would be at least 30% less in volume: because a new accent phrase is chosen only when the subdivision phrase runs out, an accent pattern will remain consistent for its entire length. The accent phrase is wholly independent from the rhythmic quantization.

Ensemble Timing

Several independent lines have the potential of sharing the same pulse in order to juxtapose separate rhythms. As a result, the parts coincide at these pulses, yet differ in their subdivisions. Random percentage variations are placed on all rhythms between pulses in order to create an elasticity of rhythm. This is an attempt to model ensemble timings in which beat is constant but the actual rhythms performed by the musicians may be anticipated or delayed.

Improvisation and Variation

A system for gesture transformation exists based on the
same principle of contour. Contours are drawn for pitch, volume, and time transformations, and act as percentage variations. Each transform contour is considered as a potential variant of 0-200% contours of 100% cause data in the corresponding contour to be considered at 100% of its value, and thus no change. Points in a transform contour of less than 100% thereby lessen the original data, whereas transform contour points greater than 100% raise original data.

Transform contours can be considered separate patterns, or can be linked to the contour they are transforming. In the latter case, the transform will affect the contour on each repeat, while in the former case, it will transform at its own independent rate. Loop points, pattern lengths, and percentage variations for both can be set, thus creating a completely independent variation gesture for pitch, volume, and time.

Events and Performance Interaction
All material is organized in terms of events, or, in the case of our performat model, hands. Combinations of pitch, velocity, and rhythm contours, when coupled with transforms and quantizations, represents a single, monophonic voice. Specific combinations of contours can be stored as patches. At the moment, sixty-four different patches can be created, recalled during performance, or saved to disk, an arbitrary limit that has proved adequate for the time being. Up to sixteen events may perform at any one time. In performance, patches can be called up as a whole, or individual contours, quantizations, and transformations can be matched separately on the Perform screen. This allows for rapid alteration of gesture in real-time.

References


MIDI 1.0 Specification 1983