Digital adaptions of the scores for Cage Variations I, II and III

Vickery
*Edith Cowan University*

Catherine Hope
*Edith Cowan University*

Stuart James
*Edith Cowan University*

Follow this and additional works at: [https://ro.ecu.edu.au/ecuworks2012](https://ro.ecu.edu.au/ecuworks2012)

This Conference Proceeding is posted at Research Online. [https://ro.ecu.edu.au/ecuworks2012/165](https://ro.ecu.edu.au/ecuworks2012/165)
Digital realizations of the scores for Cage Variations I, II and III
Lindsay Vickery, Cat Hope, and Stuart James
Western Australian Academy of Performing Arts, Edith Cowan University

ABSTRACT

Western Australian new music ensemble Decibel have devised a software-based tool for creating realizations of the score for John Cage’s Variations I and II. In these works Cage had used multiple transparent plastic sheets with various forms of graphical notation, that were capable of independent positioning in respect to one another, to create specifications for the multiple unique instantiations of these works. The digital versions allow for real-time generation of the specifications of each work, quasi-infinite exploration of diverse realizations of the works and transcription of the data created using Cage’s methodologies into proportionally rotated scrolling graphical scores.

1. INTRODUCTION

John Cage’s eight Variations (1958-67) occupy a unique position in his composer’s output. By the late 1950s, Cage had made significant progress in exploring the use of indeterminate sound sources (such as radio and LP recordings), a range of chance procedures for generating notation and indeterminacy of notation. His attention now turned towards the indeterminacy and “flexibility” of formal structure itself: “a way to further the diversity and flexibility of his compositions by removing the fixity of the score itself” [28].

The eight Variations were the principal vehicle for the exploration of this idea, constituting nearly a quarter of his compositional output during this period. Following the completion of Variations VIII, the most open of the works in every respect, Cage returned, for the most part, to more traditional compositional outcomes marked by his exploration of the “recomposition” of pre-existing works.

Over the ten years from 1958 to 1967, Cage revisited to the Variations series as a means of expanding his investigation not only of nonlinear interaction with the score but also of instrumentation, sonic materials, the performance space and the environment. The works chart an evolution from the “personal” sound-world of the performer and the score, to a vision potentially embracing the totality of sound on a global scale. Table 1 gives a summary of the evolution of Cage’s approach to the score, sound sources and the performance space in the Variations series.

<table>
<thead>
<tr>
<th>Score specification</th>
<th>sound sources</th>
<th>performance space</th>
</tr>
</thead>
<tbody>
<tr>
<td>I (1958)</td>
<td>quasi-</td>
<td>unspecified</td>
</tr>
<tr>
<td></td>
<td>determinate</td>
<td></td>
</tr>
<tr>
<td>II (1961)</td>
<td>sound</td>
<td></td>
</tr>
<tr>
<td></td>
<td>producing</td>
<td></td>
</tr>
<tr>
<td></td>
<td>means</td>
<td></td>
</tr>
<tr>
<td>III (1963)</td>
<td>indeterminate</td>
<td></td>
</tr>
<tr>
<td></td>
<td>score</td>
<td></td>
</tr>
<tr>
<td>IV (1963)</td>
<td>topographical</td>
<td></td>
</tr>
<tr>
<td></td>
<td>map</td>
<td></td>
</tr>
<tr>
<td>V (1965)</td>
<td>astronomical</td>
<td></td>
</tr>
<tr>
<td></td>
<td>chart³</td>
<td></td>
</tr>
<tr>
<td>VI (1966)</td>
<td>sound system</td>
<td>integrated</td>
</tr>
<tr>
<td></td>
<td>component</td>
<td></td>
</tr>
<tr>
<td></td>
<td>diagram</td>
<td></td>
</tr>
<tr>
<td>VIII (1967)</td>
<td>“silence”</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(ambient</td>
<td></td>
</tr>
<tr>
<td></td>
<td>sounds</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: A summary of Variations I to VIII.

Although Decibel created digital versions of Variations I-VI, this paper focuses upon the digital realization of Variations I, II and III, works that employ multiple transparent plastic sheets inscribed with either points lines or circles, for the purpose of creating a unique score for a performer to read.

1 An early example is Credo in US (1942) [34].
2 These included the use of the I Ching as a source of aleatorism in Music of Changes (1953) [29 pp. 78-88], “found systems” such as “folded paper templates” in Music for Carillon No. 1 (1952) [29 p. 92] and the “paper improvisation technique” in Music for Piano (1952-6) [29 p. 94]. Cage’s use of Astronomical maps as “found systems” dates from Music for Carillon No. 4 (1961) [29 p. 211] and was incorporated in Variations V (1965).
3 Cage’s exploration of indeterminate notation began in Music for Piano (1952) and culminated in 1958 with the magnum opus Concerto for Piano and Orchestra, [29 p. 199], [33 pp. 132, 2].
4 Examples of recomposition of pitches of Satie’s Secrètes (1918) in Cheap Imatation (1969) [4], “subtraction” of material from anthems and congregational music Apartment House 1776 (1976) and “rubbing” of Satin Chorales in Song Book (song 85) [27].
There is relatively strong documentation of the evolving non-digital performance practice of the Variations as performed by David Tudor (Variations II-1961 [30], [31], [22]), John Cage (Variations II-1963 [28]), John Cage, Merce Cunningham et al (Variations V-1965[26], [19]), David Miller (Variations I and II-2003 [24]), concerts (France), Wandel, and Tanaka (Variations VII-2008 [1]).

Traditionally, the realisation of Variations I and II in particular has necessitated time-consuming manual measurement and collation of multiple coordinates. In 2006 digital versions of Variations II were created independently by Nicholas Knouf and Pierpaolo Leo (Variations II-2006 [20], [23]). Both of these adaptations were "installation-based", in that they generated both the score and a realisation of the score for viewers to manipulate in an art gallery, rather than scored materials for live performance.

The impetus behind Decibel’s realisation of these works has been principally performative: to create practical tools for the realisation of these works that retain both the indeterminacy and the precision of the Cage’s specification.

2. VARIATIONS I AND II

In Variations I and II, Cage’s materials generate what might best be described as a blueprint for the creation of a determinate score. (Miller describes them as “toolkits” [23 p. 21].) Although Cage states that the score resulting from the application of “rules” of this work may be “simply observed” by the performer, there are significant challenges involved in actualising Variations I or II in this way (as will be discussed below).

At first glance these works appear to be a deconstruction of traditional score, with only the five stave lines and the noteheads remaining and left to float freely in two dimensions. The lines and points are in fact used by the performer to generate a unique score, in which the distance of each point from each line determines one of five musical parameters: frequency, duration, amplitude, timbre and point of occurrence.

James Pritchetti identifies the “BV” notation from Cage’s Concert for Piano (1958), illustrated in Figure 1 as the origin of this approach [29]. The connections between the “paper imperfection technique” works such as Music for Piano (1952-6), in which points representing events were spatially located on the page at knots in the surface of the paper and to and the “folded paper templates” of Music for Corillion No. 1 (1952), in which points were notated at intersections between creases in folded paper, are also significant. In Variations I the notation is, more mobile, as the lines and points are printed on transparent sheets; however the “fixes the number and structure of events” is still fixed [289 p. 136].

Earle Brown’s concept of proportional notation [18], developed some years earlier, is taken to its logical endpoint: here everything is measured. The ability to “read” the score in any orientation also draws on

Brown’s December 1952 (1954) which may be read in any direction (Left to Right, Top to Bottom, Right to Left, Bottom to Top).

The precisely defined multi-parametrical nature of Variations I also suggests the influence of the integral serial methods of the European Avant Garde, which had dominated Cage’s “chart” compositions [29 p. 78-90]. But most importantly, in these works Cage demarcates a new endpoint for the act of composition, leaving not only the interpretation, but also the final realisation of the works to the performer.

The materials for Variations I comprise six square transparencies: the first printed with points and the other five printed with lines. Square I consists of 27 points of four sizes corresponding to the number of sounds they represent as illustrated in Table 2.

<table>
<thead>
<tr>
<th>Square</th>
<th>27 Points</th>
<th>No. of Sounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>Very Small</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>Small but Larger</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Greater size</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Largest</td>
<td>4+</td>
</tr>
</tbody>
</table>

Table 2: The contents of Variations I square 1

Each of the five additional squares is printed with five lines corresponding to the five parameters shown in Table 3. The performer may freely choose which parameter to apply to each line.

<table>
<thead>
<tr>
<th>Squares 2-6</th>
<th>5 Lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>frequency</td>
</tr>
<tr>
<td>2</td>
<td>overtone structure</td>
</tr>
<tr>
<td>3</td>
<td>amplitude</td>
</tr>
<tr>
<td>4</td>
<td>duration</td>
</tr>
<tr>
<td>5</td>
<td>occurrence</td>
</tr>
</tbody>
</table>

Table 3: Variations I Squares 2-6 showing the parameters to be assigned to each line.
A reading of the work is created by measuring the distance from each point to each of the five lines to generate a composite of parameters that define each event with the following attributes: number of sounds (1-4+), frequency, duration, amplitude, timbre and point of occurrence. These attributes are relative with the continuum upon which the parameter is measured defined by the performer. For example: the point of occurrence of each event is relative to the total duration of the work (which is not defined by Cage). Figure 2 illustrates the measurement process required to define one event [16].

This procedure results in a mixture of determinate, permutable and indeterminate variables in Variations I. The number and position of the points and lines is fixed and there is a finite number of possible combinations and orientations of the transparencies, however the range of the continuum upon which each parameter is plotted is indeterminate. Table 4 illustrates the determinate, permutable and indeterminate factors involved in the generation of an instantiation of the work.

<table>
<thead>
<tr>
<th>Determinate</th>
<th>No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points/Sounds</td>
<td>27</td>
</tr>
<tr>
<td>Lines/Parameters</td>
<td>5</td>
</tr>
<tr>
<td>Min. no. of parameters</td>
<td>135 (27x5)</td>
</tr>
<tr>
<td>Permutations</td>
<td></td>
</tr>
<tr>
<td>Orientation of Points Square</td>
<td>8</td>
</tr>
<tr>
<td>Function of Lines</td>
<td>55 (120)</td>
</tr>
<tr>
<td>Orientation of Lined Squares</td>
<td>8</td>
</tr>
<tr>
<td>No. of Lined Squares</td>
<td>5</td>
</tr>
<tr>
<td>Max. No. of Permutations</td>
<td>38400</td>
</tr>
</tbody>
</table>

Variations II uses a similar system of dots and points, with some small but significant differences. There are six transparencies each with a single line and five transparencies each with a single point. The sixth line determines the structure of the musical event, whether it is a single sound, an aggregate or a constellation of sounds, the function that had been determined by the size of the points in Variations I.

The orientation of the lines and points is therefore completely open, meaning that there are an infinite set of potential configurations of the score. A performance consists of any combination of configurations and therefore in theory Variations II may describe any possible musical work [24 p. 42]. In this sense it “represents the most flexible composition tool that Cage ever invented” [29 p.136].

Performance of Variations I and II has traditionally involved one of three methods: “simply observing” [5] the resulting score, annotating an instantiation of the score [5][21] or transcribing the detailed measurements of an instantiation into a “performance score” [24 p. 22]. Figure 3 shows violinist Patricia Kopatchinskaja’s annotation of the score of Variations I [21].
The principal issue associated with “simply observing” or annotating the score, as can be observed in Figure 3, is that the notation on Cage’s transparencies is two-dimensional as opposed to traditional one-dimensional linear musical notation. To preserve the order of note occurrence, the transparencies must be read “two-dimensionally” in arcs emanating from the line that determines “point of occurrence” as shown in Figure 4. The distances to the other four lines and calculation of their parametrical value must occur simultaneously.

Figure 5: David Tudor’s transcription of Variations II [30]

Although David Tudor’s realisation of Variations I relied on “careful definition of measurement scales and a precise performance score” [30 p. 2], James Pritchett shows that Tudor’s version of Variations II reduced Cage’s prescribed measurements to binary values: simple and complex. Figure 5 shows Tudor’s transcription of two events from the work. Tudor’s score overcomes the issue of reading multiple axes (the 50 events he used were aligned in rows), however its transformation of the multi-parametrical notation into single- or double-bordered squares with intersecting lines and circled or plain points is nearly as enigmatic looking as the original.

3. THE SCORE-READER FOR VARIATIONS I AND II

The imperative of generating performance materials that are easily and intuitively read, led Decibel to a decision to transcribe the data created in Variations I and II, into proportionally notated graphical scores. In Decibel’s realisations of Variations I and II the parametrical data derived from measuring perpendicular distances is evaluated and then used to generate a scrolling, proportionally notated screen-score. The score moves from right to left with the point of occurrence of each event, rendered as a horizontal rectangle, indicated by its point of contact with a vertical line or “play-head” on the left of the screen. In this way the score moves “towards” the performer from the right in the same direction as a traditional paper score.

Duration is represented proportionally by the length of the rectangle. The vertical position of the rectangle indicates its frequency, thickness indicates volume and shade indicates timbre. The number of sounds in each event is specified by a number attached to each rectangle. A portion of such a realisation is shown in Figure 6. The notation draws on conventions established in works by Cage and his colleagues Earle Brown and Christian Wolff, as illustrated in Figure 7.

Figure 6: Decibel’s scrolling, proportionally notated screen-score for Variations I. The arrow indicates the direction of the scrolling score.

Figure 7: Graphical Notation Conventions drawn from a.) Cage Aria (1958) [8] - Timbre-Shade equivalence; b.) Wolff Duo for Pianists I (1957) [35] - numbers representing the number of sounds in an event; c.) Brown Folio and 4 systems (1954) [5] - Proportional Notation: length-duration and thickness-amplitude equivalence [2], [17], [31], [32]
This transcription enables a faster reading for performers using familiar symbols. In addition, the graphics appear ahead of the ‘playhead’, giving them time to consider their approach to the graphics. The score can be read by a single performer or multiple performers.

A control panel (shown in Figure 8) allows for determination of the total duration of the piece as well as the relative duration of the events. The range of the continuums of frequency, amplitude and timbre indicated by the score is interpreted by the performer(s) on their own instrument. The duration of the work effects the density of events on the score; for example a duration of 360 seconds will distribute the 27 events over six minutes.

![Figure 8: The master control panel for Dechel’s realisation of the Cage Variations.](image)

<table>
<thead>
<tr>
<th>Variations 1</th>
<th>Variations 2</th>
<th>Variations 3</th>
<th>Variations 4</th>
<th>Variations 5</th>
<th>Variations 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score Player</td>
<td>Score Player</td>
<td>Score Player</td>
<td>Score Player</td>
<td>Score Player</td>
<td>Score Player</td>
</tr>
<tr>
<td>Note Event</td>
<td>Note Event</td>
<td>Note Event</td>
<td>Note Event</td>
<td>Note Event</td>
<td>Note Event</td>
</tr>
<tr>
<td>-260 duration</td>
<td>-260 duration</td>
<td>-260 duration</td>
<td>-260 duration</td>
<td>-260 duration</td>
<td>-260 duration</td>
</tr>
<tr>
<td>-100 Note Duration</td>
<td>-100 Note Duration</td>
<td>-100 Note Duration</td>
<td>-100 Note Duration</td>
<td>-100 Note Duration</td>
<td>-100 Note Duration</td>
</tr>
</tbody>
</table>

The evaluation of the data to generate the scores of Variations I, II and III and a component of the score player were written in Java and embedded within the Max/MSP patch. The Java code for Variations I and II and the score player mechanism were written by Stuart James, and the Java code for Variations III was written by Aaron Wyatt.

There were several advantages for re-implementing these processes in Java. One of them was the ability to access the same memory space that Max/MSP is pointing to, namely Jitter matrices, by utilizing the Java Jitter API. This marrying of both Java and Jitter processing proved to be an efficient way of accumulating, storing, and sorting tables of values required for building note events in Variations I and II. For example here we see values stored into a Jitter matrix that are generated repeatedly in an outer section of code:

```java
int z = 0;
for (z = 0; z < coord.length; z++) {
    gB = x;
    storage.setcell[gB, 0, coord[z], 0];
}
```

And here we see the score player mechanism referencing a stored Jitter matrix of note values determining the note polyphony within a designated time frame:

```java
int theGroup = 0;
int group = -1;
int i = 0;
val = timescaleValues(i - 1);
if (i > timescaleValues[i])
    j = timescaleValues[i];
for (az = (i + 1); (i < timescaleValues[i]; az++)
    if (value < timescale)
        grouped = grouped; +1;
    else if (value > timescale)
        break;
}
```

4. Variations III

In Variations III, Cage moved to a significantly different score paradigm. Here the composer’s focus was on actions rather than sounds. The score is created by distributing 40 circles (pointed on individual transparencies) onto a surface and then removing all but the largest group of circles that are in direct contact with one another. According to Fitchett, Cage’s aim was to “enable free and direct action in the performance – one would simply do things and count the actions and variables in performance” [29 p. 149].

The digital screen score for Variations III mimics this procedure: first randomly distributing circles on the screen, then calculating the distances between them and fading out all but the largest group of overlapping circles.

7 The actions could not include those that result in an instrumental performance. Cage’s 1943 performance of Variations III included rearranging electrical cords, putting on his glasses, smoking a cigarette, writing a letter, and shaking a glass of water” [29 p. 149].
For this purpose there were advantages for implementation in a procedural language like Java by making use of recursive function calls. This proved to be significantly faster to process the way than using a scheduled message environment like Max. The calculations were made in the following way:

```java
for (int i = 0; i < circle.length; i++) {
    double distance = Math.abs(circle[i].getX() - circle[j].getX()) + 
                    Math.abs(circle[i].getY() - circle[j].getY());
    if (distance > r1 + r2) { // If distance is greater than the sum of radii
        // Then circle intersects
        // intersections with other circles in order to create a linear horizontal score. However, whereas the materials of Variations I and II give rise to a linear score, Variations III specifically evokes the indeterminacy of the two-dimensional score itself. Cage instructs the performer to "start at any circle" and "move on to any circle" only requiring the performer to "observe the number of circles which overlap it" [16]. This realization then, simply provides the means to (very rapidly) regenerate the score until they are content with the resulting graphic, while retaining the indeterminacy intrinsic to the Cage aesthetic of the original score.

4 CONCLUSION

Rendering these works digitally amounts to a procedurally specified algorithm that can call against four randomly instantiated works such as the Variations I, II, and III. On one hand, since the audience always hears the works in a linear fashion sequentially, in time there is always the question of the indeterminacy in somehow "faking" that the performance changed or before hand. In addition, such works are sometimes viewed in the audience as "mocking" the audience. As such, the mimics that the performers are themselves "mocking it up" because there is no way to determine whether they are actually reading the score.

This precision provided by the computer-plays for Variations I and II, arguably lends legitimacy to the performance, because the score that is created is both "accurate" to a reasonable degree and easily read by the performer in an audible manner.

On the other hand, such works are sometimes criticized on the grounds that potential executions of other versions imply that the particular one that is being performed might not be the best example of the work. The ability to almost instantaneously generate multiple versions of the work as demonstrated in Variations II provides the opportunity to choose interesting and promising instantiations of the work.

We have attempted to be as authentic as possible to the specifications Cage prescribed in these three works, using technology to provide a platform that is precise and accurate in its realization while still leaving open the human element of interaction with the score. As Miller expresses in regard to "authenticity" in the performance of these works:

Cage's formal statements [...] should be taken as points of departure and of periodic return in the course of developing mechanisms. They are the documents that express, however uniquely of times, the works' potentials and pre-stations [25, p. 6].
ABSTRACT

The number of solutions involved in many algorithmic composition problems is too large to be tractable without simplification. Given this, it is critical that composition algorithms be able to move through different levels of abstraction while maintaining a well-organized solution space. In this paper we present the following contributions: (1) extended formalizations and proofs needed to implement the chord spaces defined by Tymoczko [11] and Callender et al. [2], (2) a generalized framework for moving between levels of abstraction using quotient spaces that can easily be integrated with existing algorithmic composition algorithms, and (3) an application of both to voice-leading assignment.

1. INTRODUCTION

A major problem in the area of algorithmic composition is the need for organized and easily traversable sets of solutions, also referred to as solution spaces, which are tractable in terms of both runtime and memory requirements. Many music-theoretic ideas are also not formalized to the degree necessary to ensure correct implementation of algorithms and accompanying data structures. In this paper we address both of these problems by presenting a general framework for organizing and traversing harmony-related solution spaces. Our work builds on that of Tymoczko [11] and Callender et al. [2]. A generalized framework for moving between levels of abstraction using quotient spaces that can easily be integrated with existing algorithmic composition algorithms, and an application of both to voice-leading assignment.

I. INTRODUCTION

A major problem in the area of algorithmic composition is the need for organized and easily traversable sets of solutions, also referred to as solution spaces, which are tractable in terms of both runtime and memory requirements. Many music-theoretic ideas are also not formalized to the degree necessary to ensure correct implementation of algorithms and accompanying data structures. In this paper we address both of these problems by presenting a general framework for organizing and traversing harmony-related solution spaces. Our work builds on that of Tymoczko [11] and Callender et al. [2]. A generalized framework for moving between levels of abstraction using quotient spaces that can easily be integrated with existing algorithmic composition algorithms, and an application of both to voice-leading assignment.

1. INTRODUCTION

A major problem in the area of algorithmic composition is the need for organized and easily traversable sets of solutions, also referred to as solution spaces, which are tractable in terms of both runtime and memory requirements. Many music-theoretic ideas are also not formalized to the degree necessary to ensure correct implementation of algorithms and accompanying data structures. In this paper we address both of these problems by presenting a general framework for organizing and traversing harmony-related solution spaces. Our work builds on that of Tymoczko [11] and Callender et al. [2]. A generalized framework for moving between levels of abstraction using quotient spaces that can easily be integrated with existing algorithmic composition algorithms, and an application of both to voice-leading assignment.

I. INTRODUCTION

A major problem in the area of algorithmic composition is the need for organized and easily traversable sets of solutions, also referred to as solution spaces, which are tractable in terms of both runtime and memory requirements. Many music-theoretic ideas are also not formalized to the degree necessary to ensure correct implementation of algorithms and accompanying data structures. In this paper we address both of these problems by presenting a general framework for organizing and traversing harmony-related solution spaces. Our work builds on that of Tymoczko [11] and Callender et al. [2]. A generalized framework for moving between levels of abstraction using quotient spaces that can easily be integrated with existing algorithmic composition algorithms, and an application of both to voice-leading assignment.

I. INTRODUCTION

A major problem in the area of algorithmic composition is the need for organized and easily traversable sets of solutions, also referred to as solution spaces, which are tractable in terms of both runtime and memory requirements. Many music-theoretic ideas are also not formalized to the degree necessary to ensure correct implementation of algorithms and accompanying data structures. In this paper we address both of these problems by presenting a general framework for organizing and traversing harmony-related solution spaces. Our work builds on that of Tymoczko [11] and Callender et al. [2]. A generalized framework for moving between levels of abstraction using quotient spaces that can easily be integrated with existing algorithmic composition algorithms, and an application of both to voice-leading assignment.

I. INTRODUCTION

A major problem in the area of algorithmic composition is the need for organized and easily traversable sets of solutions, also referred to as solution spaces, which are tractable in terms of both runtime and memory requirements. Many music-theoretic ideas are also not formalized to the degree necessary to ensure correct implementation of algorithms and accompanying data structures. In this paper we address both of these problems by presenting a general framework for organizing and traversing harmony-related solution spaces. Our work builds on that of Tymoczko [11] and Callender et al. [2]. A generalized framework for moving between levels of abstraction using quotient spaces that can easily be integrated with existing algorithmic composition algorithms, and an application of both to voice-leading assignment.