Techniques for Algorithmic Composition, part II

A brief Treatise outlining
processes for manipulating
melodic, harmonic and rhythmic progressions

By David Matthew Shere

TECHNIQUES for ALGORITHMIC COMPOSITION part II

by David M. Shere; copyright © 2006

TABLE OF CONTENTS

I. Introduction

This paper is an extension of a previous treatise which was completed in December 2004. It is the second part in a series which may eventually become a full-length textbook. The topics contained herein represent the continuing evolution of certain algorithmic techniques which I use for music composition. A number of ideas discussed in this paper are previously undescribed, while others constitute the further development of algorithms introduced in the first paper.

While I consider these techniques to be unique or unusual enough to warrant further study and exploration, I do not claim that these are completely novel ideas from a theoretical standpoint. Each technique that I will be discussing here can generally be shown to have a precedent based on the work of another composer. Wherever possible, I have made every attempt to cite the relevant works of those individuals, and show how their compositional methods provided the inspiration for these ideas.

-David M. Shere
January 22, 2006
Santa Barbara, CA

II. A Brief Explanation of the Format

As this is the first draft of this treatise, each key concept will be explained as concisely as possible, and briefly illustrated with accompanying figures and/or musical examples from my own work. The verbal explanations comprise the main body of the paper; the musical examples can be found in the first appendix. The second appendix contains a template of a checklist of chromatic aggregates; the third appendix contains the bibliography.

With regard to the ordering of topics: Each concept has been presented more or less in order of historical precedent, evolution and degree of complexity, beginning with topics that apply to neo-tonality and centricity, and progressing to more complex dodecaphonic procedures. I have also made every effort to distinguish, by order of presentation and sub-grouping, those topics which apply to manipulating **harmony** from those topics which apply to manipulating **melody**; however, there is a certain amount of overlap between the two ideas, and it is not possible to *completely* separate these topics by this distinction. Processes which apply to manipulating harmony may also be applied to manipulation of melody, and vice versa.

III. Algorithms for Manipulating Harmony

1. Pitch-relative Harmony

In the context of post-modern music and the *avant-garde*, the technique of constructing a harmonic progression is no longer bound by any common-practise set of rules whatsoever. Consequently, harmonizing a given melody has become an exercise in truly arbitrary decision-making. This technique can be found to some degree in the harmonic language of every known composer from the late Romantic period until the present day. While it is not a novel idea, it bears a certain amount of description in order to lay the foundation for certain techniques which will follow it in this paper.

- Application

A single pitch can be harmonized by any triad, 7th chord, embellished chord, or chromatic pitch-set one chooses **[example 1-1]**. When a single pitch is used as a centric point around which all of the harmony in a composition is organized, the contemporary term for this organizational scheme is **"pitch-axis harmony"**[1]. In and of itself, the pitch-axis technique is somewhat limited, and ultimately is not exceptionally interesting. However, if we take a given melody **[example 1-2]** and treat *every pitch* in that melody as a harmonic axis **[example 1-3]**, generating a series of variations on the melody in which the harmony for each pitch-class changes with every variation **[example 1-4]**, we can achieve truly interesting compositional results.

- Analysis and harmonic function

In the same way that the choice of harmony for any pitch-class has become arbitrary, the analysis of that harmony in context can also be viewed as somewhat arbitrary. The harmonic nomenclature becomes dependent on 1) what system of analysis you choose to use (Roman numeral, jazz chord symbol, set theory, etc.), and 2) what you choose to view as the root of the harmony. Several examples showing multiple analyses of a single harmonic setting can be found in the appendix **[example 1-5]**.

---

[1] Marshall, Wolf. **Joe Satriani: Surfing with the Alien.** "Introduction." Cherry Lane Music Company, Inc.: Port Chester NY, 1987.

2. Rotational Pentachords

Any 12-tone row can be divided into two hexachords [**example 2-1**], which can then be treated as verticalities. If we place these two hexachords side-by-side, the first *descending* vertically and the second *ascending* vertically [**example 2-2**], the row takes on the visual characteristic of having been "wrapped around" a vertical axis. We can then manipulate the row according to this characteristic.

If we "rotate" the row 5 times around this imaginary vertical axis for a total of six pairs of vertical hexachords [**example 2-3**], an extremely interesting pattern emerges. Notice the numerical relationship that emerges between the top two rows, where pairs of numbers manifest a consistent diagonal relationship with one another. From a pitch-class perspective, this amounts to a continuous voice-exchange between these two upper row positions.

If we eliminate one of these top two rows [**example 2-4**], we are left with a peculiar series of 12 vertical pentachords comprised of 5 independent horizontal rows, in which every pitch-class is displaced from all other iterations of that pitch-class by the maximum possible horizontal distance. Furthermore, no two vertical pentachords in this series contain exactly the same pitch-class content; each vertical sonority is unique and distinct.

The compositional possibilities of such a pentachord series are endless. This technique has the advantage of generating rich, interesting harmonies while avoiding undue pitch-class repetition or centricity, and without violating the fundamental serial aesthetic.

3. Encrypted Trichordal arrays

- A brief review of sequence filtering, or **serial encryption**

A 12-tone row can be manipulated in a cryptographic manner similar to any type of information which has been stored in an **array**. Each **element** in the array can be assigned an **index** value. Index patterns may then be created, and the positions of the elements in the array may be manipulated according to these patterns **[example 3-1]**. This index-pattern manipulation process is essentially a rudimentary form of **encryption**.

- Creating asymmetrical trichordal arrays

Any 12-tone row may be used to create a trichordal array, by rotating the original row four times and "stacking" the results to create tetrachordal verticalities **[example 3-2]**. Any trichordal array which is "asymmetric" does not conform to the typical definition of trichordal arrays, which requires that all four trichords in a row should have the same prime form.

- Purpose of encryption

The drawback of any typical trichordal array (whether symmetrical or asymmetrical) is that it limits the practitioner to exactly four discrete horizontal trichords and three distinct vertical tetrachords derived from any given row. While the interval content of each tetrachord verticality varies somewhat, the pitch-class content does not **[example 3-3]**. The horizontal trichord content is typically fixed and does not vary.

In order to derive greater harmonic and melodic variety from a single row, it is possible to apply serial encryption methods to a trichordal array. This process can essentially be described as filtering the pitch-class and interval-content identity of one row through the index-sequence identity of another row, resulting in an **encrypted trichordal array**.

- Description of the encryption process

Each vertical tetrachord may be treated as an element in an array and assigned an index value **[example 3-4]**. An index pattern may then be created **[example 3-5]** with which to manipulate (or encrypt) the positions of the elements in the array. In my own work, I construct these index patterns according to specific guidelines which prevent iterations of the same pitch-class in close proximity with each other. Essentially, every tetrachord belongs to one of three "families", and its position may be "swapped" (interchanged) with any other member of that family, without affecting the proximity of any pitch-class with other iterations of that pitch-class within the trichordal matrix. Thus, each index position may also be associated with one of three families **[example 3-6]**, and index patterns may be constructed by cycling through the contents of each family systematically **[example 3-7]**, essentially interchanging index positions which are related to one another.

Each transposition of the of the index pattern **[example 3-8]** may then also be used as an encryption filter (as well as the retrograde, inversion, and retrograde-inversions of the index pattern and their transpositions, if the practitioner wishes). The resulting encrypted arrays **[example 3-9]** can each clearly be seen to have a new melodic profile for each component row. This process defeats the original limitation of four discrete horizontal trichords.

The identity of each vertical tetrachord verticality remains intact; however, this limitation may also be defeated. By transposing dichords within the trichordal matrix **[example 3-10]**, rotating a single row one place to the left or right, or some other additional method, the result is 12 distinct vertical tetrachords, no two of which have exactly the same pitch-class content. Basically, by using encryption one may derive 48 distinct horizontal rows and 144 distinct vertical tetrachords from a single twelve-tone row.

4. 12-tone Chordal counterpoint

Interesting vertical sonorities can be constructed using all 12 pitch-classes **[example 4-1]**. These types of harmonies are evident in the works of composer Witold Lutoslawski, among others, and are also thoroughly discussed in the **Thesaurus of Scales and Melodic Patterns** by Nicholas Slonimsky.

Any given 12-tone chord may then be used as a basis for counterpoint, by applying voice-exchanges from one sonority to the next to create a 12-tone chord progression **[example 4-2]**. While the goal of this technique is to have all 12 pitch-classes present in every consecutive harmony, the nature of each sonority changes significantly with successive manipulations of the interval content.

IV. Algorithms for Manipulating Melody

5. 12-tone scales

An interesting application of the serial technique is to create scale-like structures which contain all 12 pitch classes. Included in the appendix are a few examples of possible 12-tone "scales" **[example 5-1]**. The compositional interest of these constructs is that they permit a relatively greater degree of idiomatic application and traditional figuration writing than typical 12-tone arrays, due to their incorporation of predominantly smaller intervals arranged in step-wise fashion in a single direction.

6. Index Subset Permutations of 12-tone rows

Within a given 12-tone row, the contents of smaller subsets of elements can be manipulated. The index order of these subsets may be interchanged, creating melodic variations on that row which bear a strong resemblance to the original row **[example 6-1]**. This technique is useful for deriving melodic and harmonic variety from a single row, without introducing completely new and unrelated rows to the compositional mix. The question this process raises is, "How many index changes can a row sustain before it has completely lost its individual identity as a series?" At this time, the answer to this question is unclear.

V. Algorithms for Manipulating Rhythm

7. Rotational rhythm arrays

In the same manner that a fixed series of pitches may be contrived and used as a compositional device, it is also possible to contrive a fixed series of rhythmic values. This is essentially a type of **integral serialism**, as exhibited in the works of composers such as Olivier Messiaen, Pierre Boulez, Karl Stockhausen, Milton Babbitt, and others[2].

In my own work, I have used such a rhythmic series as a **rotational array**, in conjunction with a pitch series **[example 7-1]**. Initially, each rhythmic value in a table

---

[2] Chung , Ki Hyang. **Integral Serialism and the Rise of the International Avant-garde.** 1/31/06. <http://www.usc.edu/dept/polish_music/578/aug05.html>

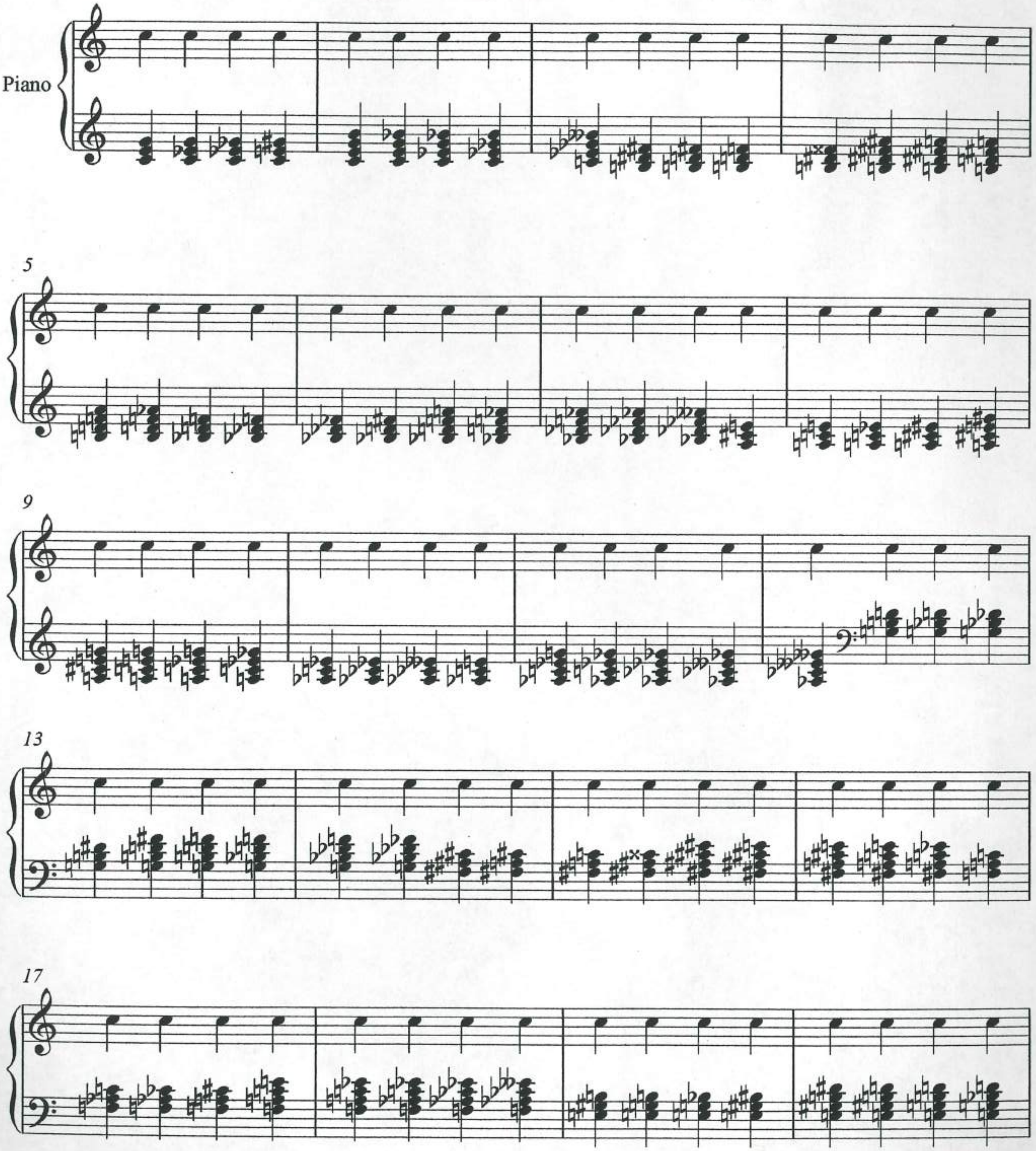
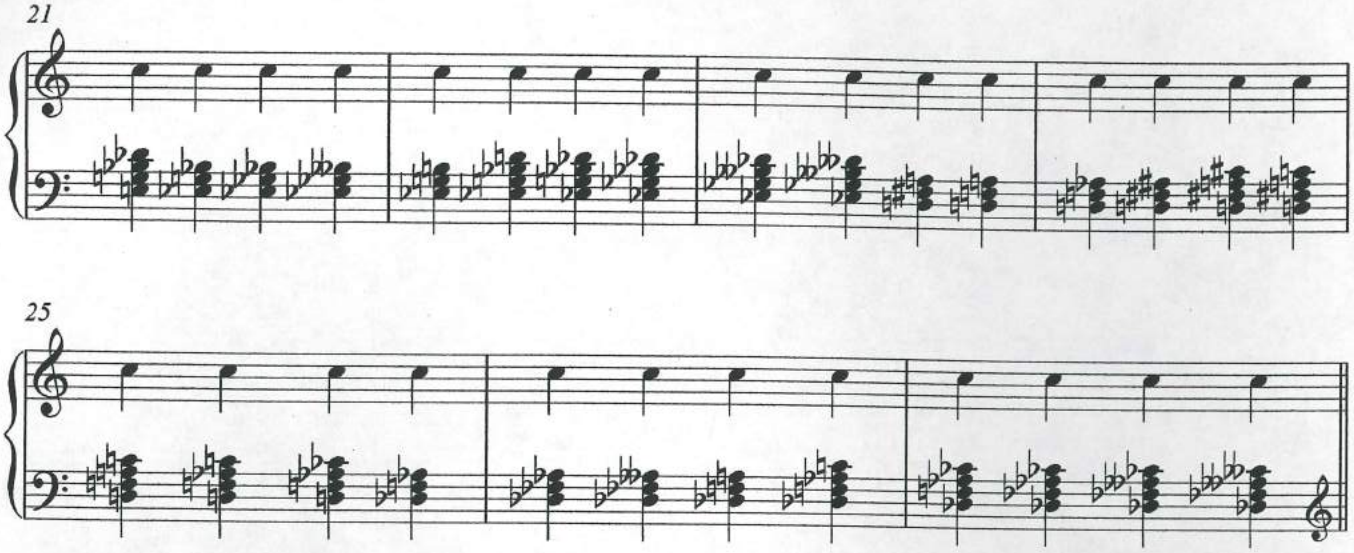
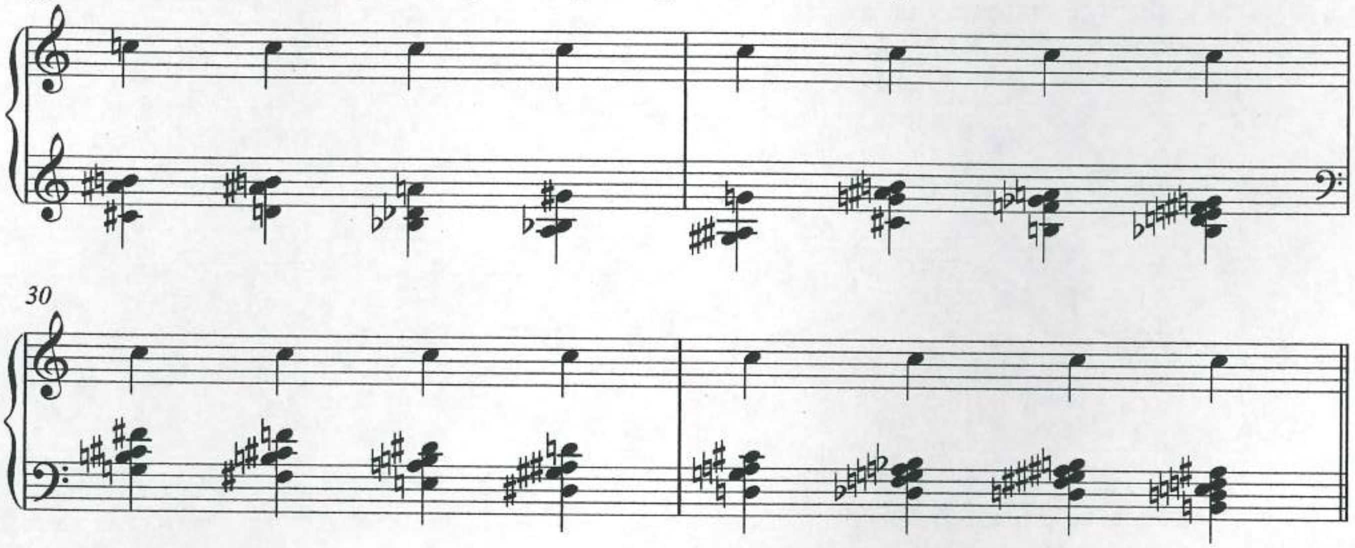
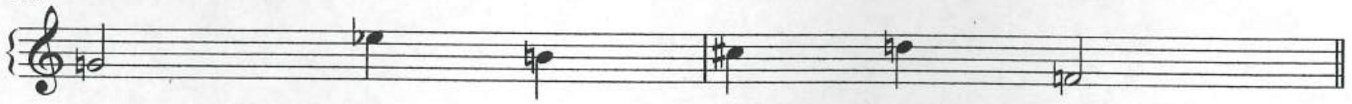
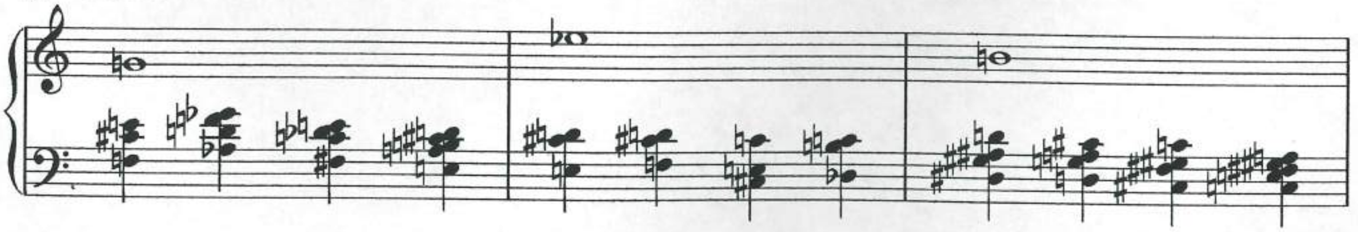
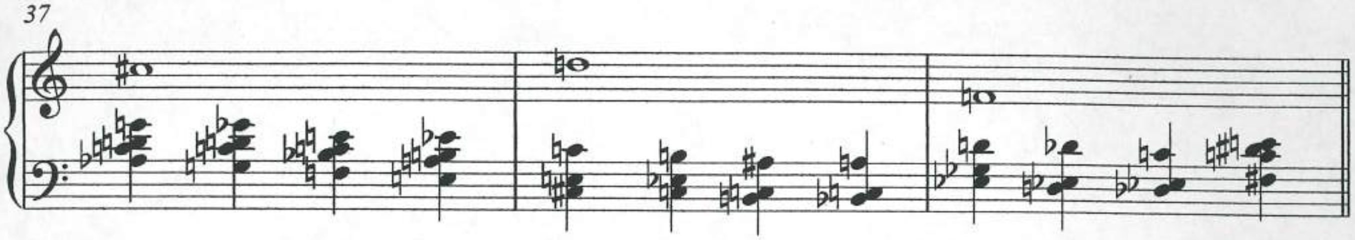
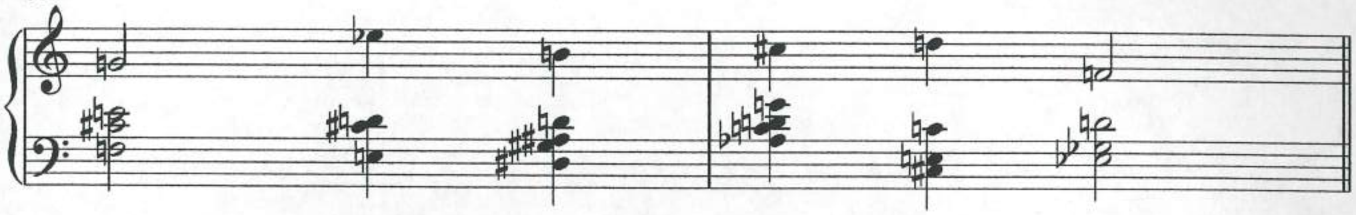
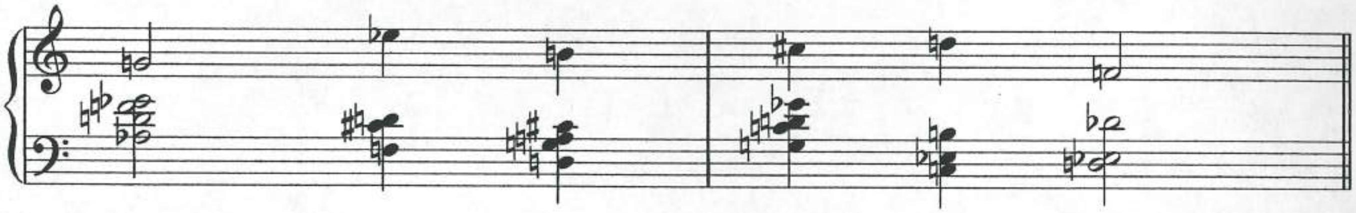
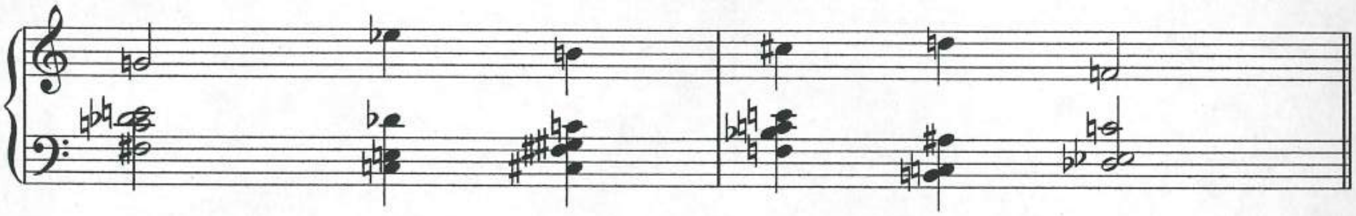
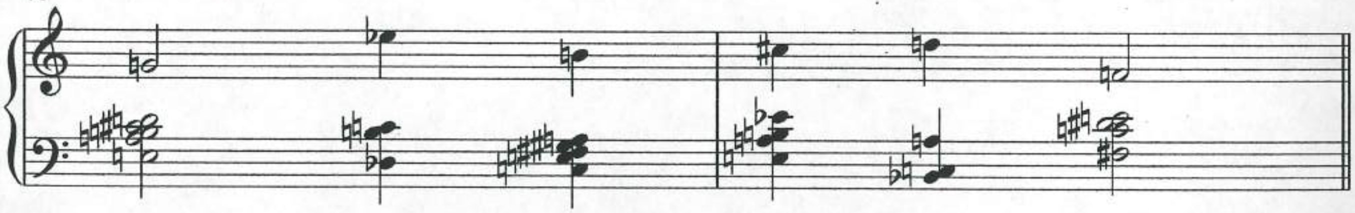
corresponds to a specific pitch-class value. As the rhythm array is rotated through the table, the corresponding pitch-class values change accordingly, permitting every pitch-class to correspond with every rhythm value in the array by the end of the process.

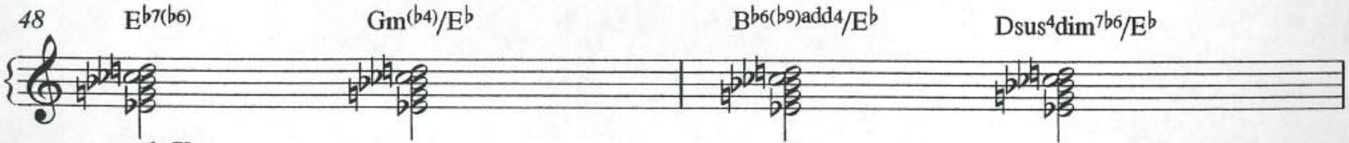
There are endless possibilities for constructing a rhythm array. In my own work, I "weight" my arrays according to what types of rhythmic values I wish to emphasize. Included in the appendix are several more examples of possible rhythm arrays, demonstrating weighted use of one or more rhythmic denominations [**example 7-2**].

Appendix 1
Musical Examples

David M. Shere

♩ = 96

ex. 1-1a (harmonizing a single pitch using triads/7th chords in all 12 keys)

ex. 1-1b (harmonizing a single pitch using a sample of chromatic sets)

ex. 1-2

ex. 1-3

ex. 1-4a

ex. 1-4b

ex. 1-4c

ex. 1-4d

ex. 1-5a



ex. 1-5b

(prime form = [01458]; set 5-21)

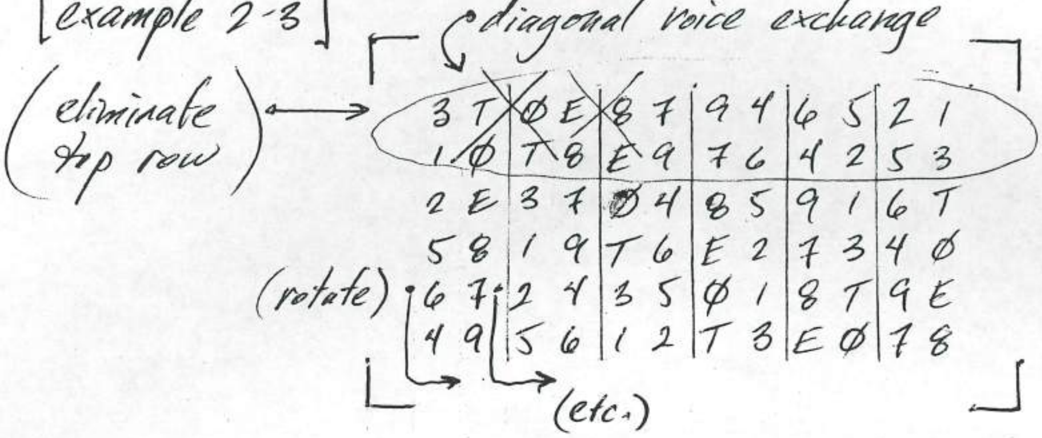
8/24/05 – 5-note chords by hexachordal rotation (Rotational pentachords):

[example 2-1] **Row:** $[\overset{H_1}{3\ 1\ 2\ 5\ 6\ 4}\ |\ \overset{H_2}{9\ 7\ 8\ E\ 0\ T}]$

[example 2-2] **HEXACHORDS:**

H_1	
3	T
1	0
2	E
5	8
6	7
4	9
	H_2

[example 2-3]

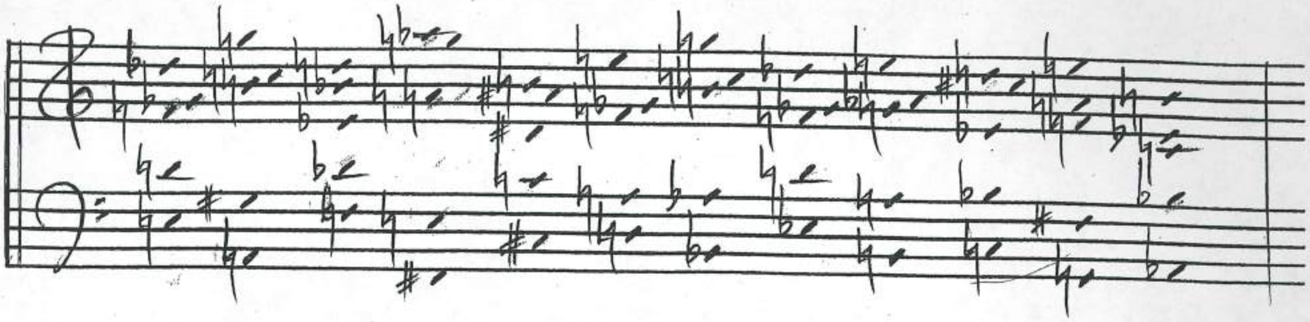


3 T	0 E	8 7	9 4	6 5	2 1
1 0	T 8	E 9	7 6	4 2	5 3
2 E	3 7	0 4	8 5	9 1	6 T
5 8	1 9	T 6	E 2	7 3	4 0
6 7	2 4	3 5	0 1	8 T	9 E
4 9	5 6	1 2	T 3	E 0	7 8

[example 2-4] final pentachord array

1 0 T 8 E 9 7 6 4 2 5 3
2 E 3 7 0 4 8 5 9 1 6 T
5 8 1 9 T 6 E 2 7 3 4 0
6 7 2 4 3 5 0 1 8 T 9 E
4 9 5 6 1 2 T 3 E 0 7 8

8/24/05 - Hexachordal rotations (resulting in pentachords)



[example 1-4]

- rendered in standard notation

2/8/06 - Review of serial encryption

[example 3-1]

Row = [E 6 7 T 8 9 1 2 5 3 4 0]

INDEX = $\overrightarrow{0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ T\ E}$

↓

ELEMENTS =

$$\begin{array}{lccc} \text{INDEX PATTERN} = & [0\ 4\ 3\ 7\ 8] & \longrightarrow & [E\ 8\ T\ 2\ 5] \\ & [1\ 5\ 4\ 8\ 9] & \longrightarrow & [6\ 9\ 8\ 5\ 3] \\ \text{(index} & [2\ 6\ 5\ 9\ T] & \longrightarrow & [7\ 1\ 9\ 3\ 4] \\ \text{sequence)} & [3\ 7\ 6\ T\ E] & \longrightarrow & [T\ 2\ 1\ 4\ 0] \\ & \ldots\ (\text{etc.}) & & \ldots\ (\text{etc.}) \\ & \downarrow & & \downarrow \end{array}$$

8/16/05 - ENCRYPTED Trichordal Matrices
Using a blues-based 12-tone row.

BASIC ROW: [E 6 7 T 8 9 1 2 5 3 4 0]

[example 3-2]

TRICHORDAL MATRIX:

$$\left[\begin{array}{c|ccc|ccc|ccc|ccc} & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & T & E \\ ROT_1 & E & 6 & 7 & T & 8 & 9 & 1 & 2 & 5 & 3 & 4 & 0 \\ ROT_3 & 1 & 2 & 5 & 3 & 4 & 0 & E & 6 & 7 & T & 8 & 9 \\ ROT_4 & T & 8 & 9 & 1 & 2 & 5 & 3 & 4 & 0 & E & 6 & 7 \\ ROT_2 & 3 & 4 & 0 & E & 6 & 7 & T & 8 & 9 & 1 & 2 & 5 \end{array}\right]$$

[example 3-3]

[example 3-4]

[example 3-5]

filter: [9 7 5 | 6 T 2 | 0 1 E | 3 4 8] (index sequence)

{example 3-6, 3-7 on following page}

$$\left[\begin{array}{c|cccccccccccc} -f_0 & 9 & 7 & 5 & 6 & T & 2 & 0 & 1 & E & 3 & 4 & 8 \\ -f_1 & T & 8 & 6 & 7 & E & 3 & 1 & 2 & 0 & 4 & 5 & 9 \\ -f_2 & E & 9 & 7 & 8 & 0 & 4 & 2 & 3 & 1 & 5 & 6 & T \\ -f_3 & 0 & T & 8 & 9 & 1 & 5 & 3 & 4 & 2 & 6 & 7 & E \\ -f_4 & 1 & E & 9 & T & 2 & 6 & 4 & 5 & 3 & 7 & 8 & 0 \\ -f_5 & 2 & 0 & T & E & 3 & 7 & 5 & 6 & 4 & 8 & 9 & 1 \\ -f_6 & 3 & 1 & E & 0 & 4 & 8 & 6 & 7 & 5 & 9 & T & 2 \\ -f_7 & 4 & 2 & 0 & 1 & 5 & 9 & 7 & 8 & 6 & T & E & 3 \\ -f_8 & 5 & 3 & 1 & 2 & 6 & T & 8 & 9 & 7 & E & 0 & 4 \\ -f_9 & 6 & 4 & 2 & 3 & 7 & E & 9 & T & 8 & 0 & 1 & 5 \\ -f_T & 7 & 5 & 3 & 4 & 8 & 0 & T & E & 9 & 1 & 2 & 6 \\ -f_E & 8 & 6 & 4 & 5 & 9 & 1 & E & 0 & T & 2 & 3 & 7 \end{array}\right]$$

[example 3-8]

filter array:

2/8/06 - ENCRYPTED Trichordal arrays (cont'd)

[example 3-6] INDEX POSITION FAMILIES:

[0 3 6 9]	[1 4 7 T]	[2 5 8 E]
E T 1 3	6 8 2 4	7 9 5 0
1 3 E T	2 4 6 8	5 0 7 9
T 1 3 E	8 2 4 6	9 5 0 7
3 E T 1	4 6 8 2	0 7 9 5

corresponding vertical tetrachords (see previous page)

[example 3-7] - the tetrachords in each family may be interchanged without disturbing the proximity of pitch-set iterations in the trichordal matrix

[3 9 6 0]

T 3 1 E
3 T E 1
1 E 3 T
E 1 T 3

(index)

3 1 2 | 9 4 5 | 6 7 8 | 0 T E

T 6 7 3 8 9 1 2 5 E 4 0
3 2 5 T 4 0 E 6 7 1 8 9
1 8 9 E 2 5 3 4 0 T 6 7
E 4 0 1 6 7 T 8 9 3 2 5

compare with original trichordal matrix [example 3-2] for further clarification

(return to prev. pg. for [example 3-8])

2) 8/16/05 - ENCRYPTED Arrays (Vln. Sonata #2 / Pno. Sonata #2) [example 3-9]
(prior to dichord transposition)

f_0 |9 7 5 6 T 2 0 1 E 3 4 8|

$\boxed{3\ 2}$ 9 | 1 4 7 | E $\boxed{6\ 0}$ | T 8 5
T 6 0 | $\boxed{E\ 8}$ 5 | 1 2 9 | 3 $\boxed{4\ 7}$
E 4 5 | 3 6 9 | $\boxed{T\ 8}$ 7 | $\boxed{1\ 2}$ 0
1 8 7 | T $\boxed{2\ 0}$ | 3 4 5 | E 6 9

f_1 |T 8 6 7 E 3 1 2 0 4 5 9|

$\boxed{4\ 5}$ 1 | 2 0 T | 6 $\boxed{7\ E}$ | 8 9 3
8 7 E | $\boxed{6\ 9}$ 3 | 2 5 1 | 4 $\boxed{0\ T}$
6 0 3 | 4 7 1 | $\boxed{8\ 9}$ T | $\boxed{2\ 5}$ E
2 9 T | 8 $\boxed{5\ E}$ | 4 0 3 | 6 7 1

f_2 |E 9 7 8 0 4 2 3 1 5 6 T|

$\boxed{0\ 3}$ 2 | 5 E 8 | 7 $\boxed{T\ 6}$ | 9 1 4
9 T 6 | $\boxed{7\ 1}$ 4 | 5 3 2 | 0 $\boxed{E\ 8}$
7 E 4 | 0 T 2 | $\boxed{9\ 1}$ 8 | $\boxed{5\ 3}$ 6
5 1 8 | 9 $\boxed{3\ 6}$ | 0 E 4 | 7 T 2

f_3 |0 T 8 9 1 5 3 4 2 6 7 E|

$\boxed{E\ 4}$ 5 | 3 6 9 | T $\boxed{8\ 7}$ | 1 2 0
1 8 7 | $\boxed{T\ 2}$ 0 | 3 4 5 | E $\boxed{6\ 9}$
T 6 0 | E 8 5 | $\boxed{1\ 2}$ 9 | $\boxed{3\ 4}$ 7
3 2 9 | 1 $\boxed{4\ 7}$ | E 6 0 | T 8 5

[example 3-10]

f_4 |1 E 9 T 2 6 4 5 3 7 8 0|

$\boxed{6\ 0}$ 3 | 4 7 1 | 8 $\boxed{9\ T}$ | 2 5 E
2 9 T | $\boxed{8\ 5}$ E | 4 0 3 | 6 $\boxed{7\ 1}$
8 7 E | 6 9 3 | $\boxed{2\ 5}$ 1 | $\boxed{4\ 0}$ T
4 5 1 | 2 $\boxed{0\ T}$ | 6 7 E | 8 9 3

f_5 |2 0 T E 3 7 5 6 4 8 9 1|

$\boxed{7\ E}$ 4 | 0 T 2 | 9 $\boxed{1\ 8}$ | 5 3 6
5 1 8 | $\boxed{9\ 3}$ 6 | 0 E 4 | 7 $\boxed{T\ 2}$
9 T 6 | 7 1 4 | $\boxed{5\ 3}$ 2 | $\boxed{0\ E}$ 8
0 3 2 | 5 $\boxed{E\ 8}$ | 7 T 6 | 9 1 4

■ = Dichords to be transposed

[example 3-9] (cont'd)

8/16/05 - ENCRYPTED Arrays (Vln. Sonata #2 / Pno. Sonata #2)
(prior to dichord transposition)

f_6 |3 1 E 0 4 8 6 7 5 9 T 2|

$\boxed{T\,6}$ 0	E 8 5	1 $\boxed{2\,9}$	3 4 7
3 2 9	$\boxed{1\,4}$ 7	E 6 0	T $\boxed{8\,5}$
1 8 7	T 2 0	$\boxed{3\,4}$ 5	$\boxed{E\,6}$ 9
E 4 5	3 $\boxed{6\,9}$	T 8 7	1 2 0

f_7 |4 2 0 1 5 9 7 8 6 T E 3|

$\boxed{8\,7}$ E	6 9 3	2 $\boxed{5\,1}$	4 0 T
4 5 1	$\boxed{2\,0}$ T	6 7 E	8 $\boxed{9\,3}$
2 9 T	8 5 E	$\boxed{4\,0}$ 3	$\boxed{6\,7}$ 1
6 0 3	4 $\boxed{7\,1}$	8 9 T	2 5 E

f_8 |5 3 1 2 6 T 8 9 7 E 0 4|

$\boxed{9\,T}$ 6	7 1 4	5 $\boxed{3\,2}$	0 E 8
0 3 2	$\boxed{5\,E}$ 8	7 T 6	9 $\boxed{1\,4}$
5 1 8	9 3 6	$\boxed{0\,E}$ 4	$\boxed{7\,T}$ 2
7 E 4	0 $\boxed{T\,2}$	9 1 8	5 3 6

f_9 |6 4 2 3 7 E 9 T 8 0 1 5|

$\boxed{1\,8}$ 7	T 2 0	3 $\boxed{4\,5}$	E 6 9
E 4 5	$\boxed{3\,6}$ 9	T 8 7	1 $\boxed{2\,0}$
3 2 9	1 4 7	$\boxed{E\,6}$ 0	$\boxed{T\,8}$ 5
T 6 0	E $\boxed{8\,5}$	1 2 9	3 4 7

f_T |7 5 3 4 8 0 T E 9 1 2 6|

$\boxed{2\,9}$ T	8 5 E	4 $\boxed{0\,3}$	6 7 1
6 0 3	$\boxed{4\,7}$ 1	8 9 T	2 $\boxed{5\,E}$
4 5 1	2 0 T	$\boxed{6\,7}$ E	$\boxed{8\,9}$ 3
8 7 E	6 $\boxed{9\,3}$	2 5 1	4 0 T

f_E |8 6 4 5 9 1 E 0 T 2 3 7|

$\boxed{5\,1}$ 8	9 3 6	0 $\boxed{E\,4}$	7 T 2
7 E 4	$\boxed{0\,T}$ 2	9 1 8	5 $\boxed{3\,6}$
0 3 2	5 E 8	$\boxed{7\,T}$ 6	$\boxed{9\,1}$ 4
9 T 6	7 $\boxed{1\,4}$	5 3 2	0 E 8

[example 3-10] (cont'd)

(fig.4) 8/14/05 - ENCRYPTED Arrays (Vln. Sonata #2 / Pno. Sonata #2)
WITH Diatord transpositions

8/20/05 retain
bassline but voice
upper tones
as necessary

f_0 ✓

	2 3 4	1 4 7	E 0 6	T 8 5
	T 6 0	8 E 5	1 2 9	3 7 4
	E 4 5	3 6 9	8 T 7	2 1 0
BASS →	1 8 7	T 0 2	3 4 5	E 6 9

f_1 ✓

	5 4 1	2 0 T	6 E 7	8 9 3
	8 7 E	9 6 3	2 5 1	4 T 0
	6 0 3	4 7 1	9 8 T	5 2 E
•	2 9 T	8 E 5	4 0 3	6 7 1

f_2 ✓

	3 0 2	5 E 8	7 6 T	9 1 4
	9 T 6	1 7 4	5 3 2	0 8 E
	7 E 4	0 T 2	1 9 8	3 5 6
•	5 1 8	9 6 3	0 E 4	7 T 2

f_3 ✓

	4 E 5	3 6 9	T 7 8	1 2 0
	1 8 7	2 T 0	3 4 5	E 9 6
	T 6 0	E 8 5	2 1 9	4 3 7
•	3 2 9	1 7 4	E 6 0	T 8 5

f_4 ✓

	0 6 3	4 7 1	8 T 9	2 5 E
	2 9 T	5 8 E	4 0 3	6 1 7
	8 7 E	6 9 3	5 2 1	0 4 T
•	4 5 1	2 T 0	6 7 E	8 9 3

f_5 ✓

	E 7 4	0 T 2	9 8 1	5 3 6
	5 1 8	3 9 6	0 E 4	7 2 T
	9 T 6	7 1 4	3 5 2	E 0 8
•	0 3 2	5 8 E	7 T 6	9 1 4

8/20/05 - ORIGINAL ROW [E 6 7 T 8 9 1 2 5 3 4 0]

"Blues" row

[example 3-10] (cont.'d)

5) 8/16/05 – ENCRYPTED Arrays (Vln. Sonata #2/Pno. Sonata #2)
WITH Dichord transpositions

f_6 ✓

6 T 0	E 8 5	1 9 2	3 4 7
3 2 9	4 1 7	E 6 0	T 5 8
1 8 7	T 2 0	4 3 5	6 E 9
E 4 5	3 9 6	T 8 7	1 2 0
' ' '	' ' '	' ' '	' ' '

f_7 ✓

7 8 E	6 9 3	2 1 5	4 0 T
4 5 1	0 2 T	6 7 E	8 3 9
2 9 T	8 5 E	0 4 3	7 6 1
6 0 3	4 1 7	8 9 T	2 5 E
' ' '	' ' '	' ' '	' '

f_8 ✓

T 9 6	7 1 4	5 2 3	0 E 8
0 3 2	E 5 8	7 T 6	9 4 1
5 1 8	9 3 6	E 0 4	T 7 2
7 E 4	0 2 T	9 1 8	5 3 6
' ' '	' ' '	' ' '	' ' '

f_9 ✓

8 1 7	T 2 0	3 5 4	E 6 9
E 4 5	6 3 9	T 8 7	1 0 2
3 2 9	1 4 7	6 E 0	8 T 5
T 6 0	E 5 8	1 2 9	3 4 7
' ' '	' ' '	' ' '	' ' '

f_T ✓

9 2 T	8 5 E	4 3 0	6 7 1
6 0 3	7 4 1	8 9 T	2 E 5
4 5 1	2 0 T	7 6 E	9 8 3
8 7 E	6 3 9	2 5 1	4 0 T
' ' '	' ' '	' ' '	' ' '

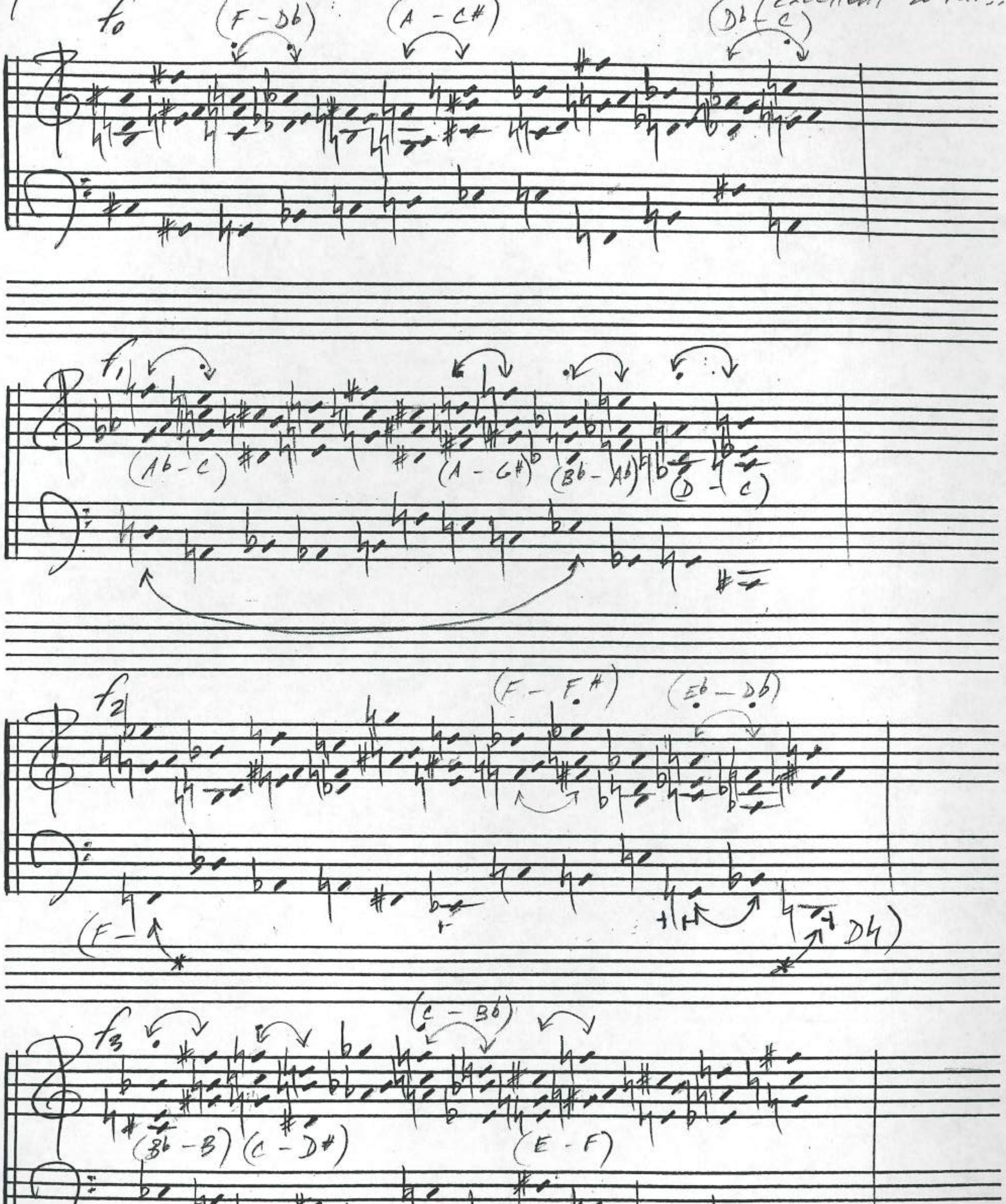
f_E

1 5 8	9 3 6	0 4 E	7 T 2
7 E 4	T 0 2	9 1 8	5 6 3
0 3 2	5 E 8	T 7 6	1 9 4
9 T 6	7 4 1	5 3 2	0 E 8
' ' '	' ' '	' ' '	' ' '

"Blues" row

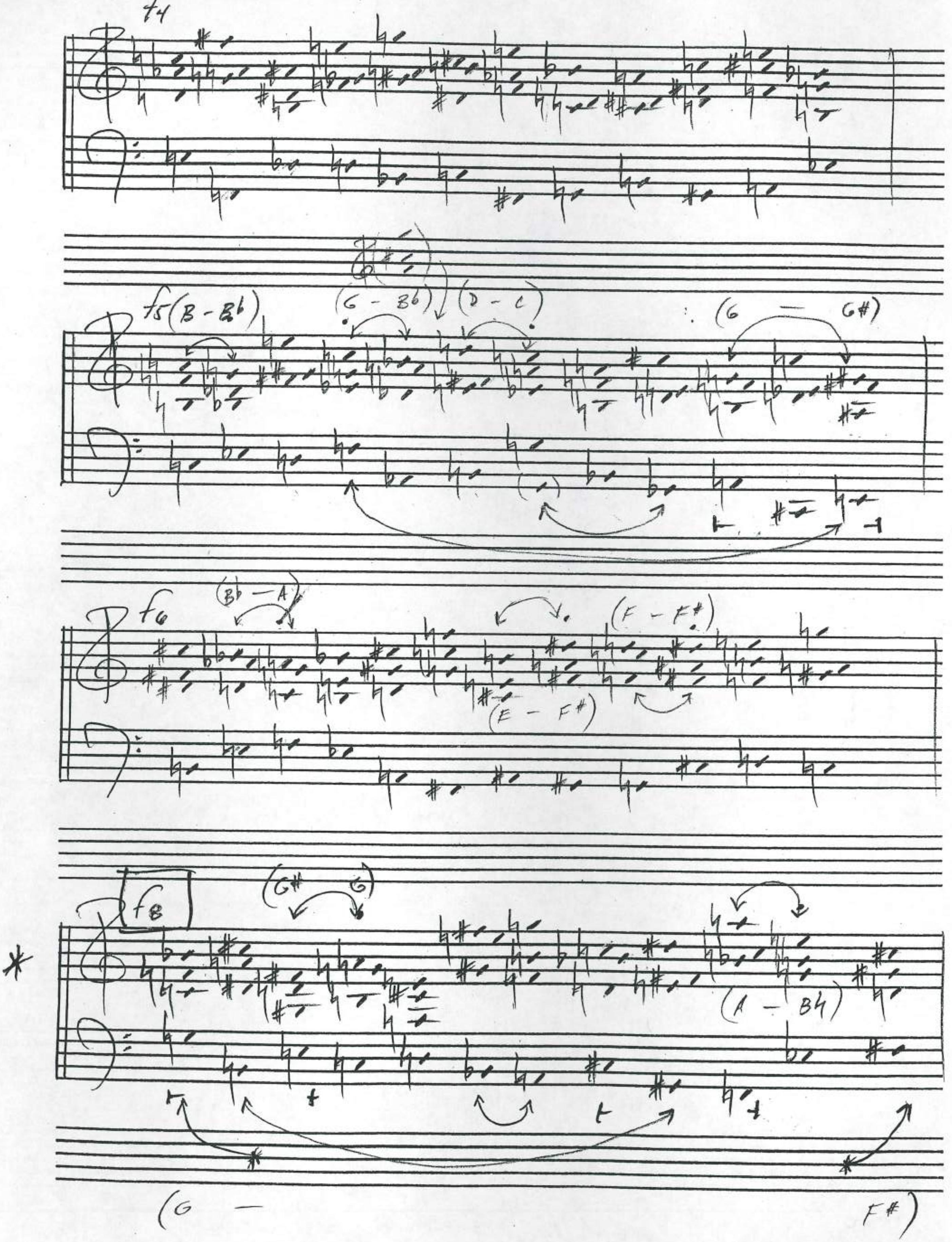
8/20/05 - ENCRYPTED Arrays (8/16/05 row)

(excellent so far...)

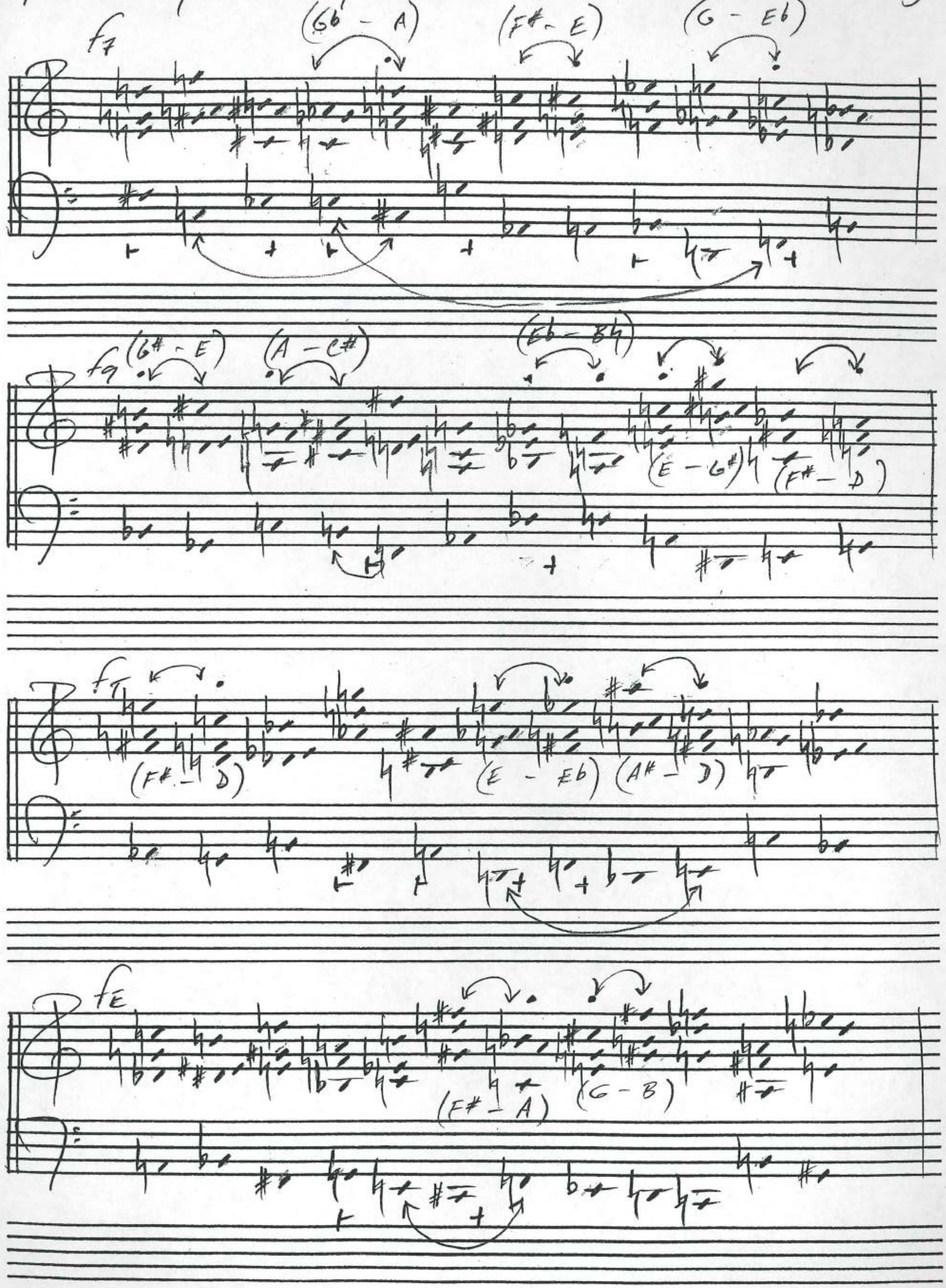


8/21/05 - Use voice exchanges and added tones to get rid of Triads; also V.E.'s in the bassline

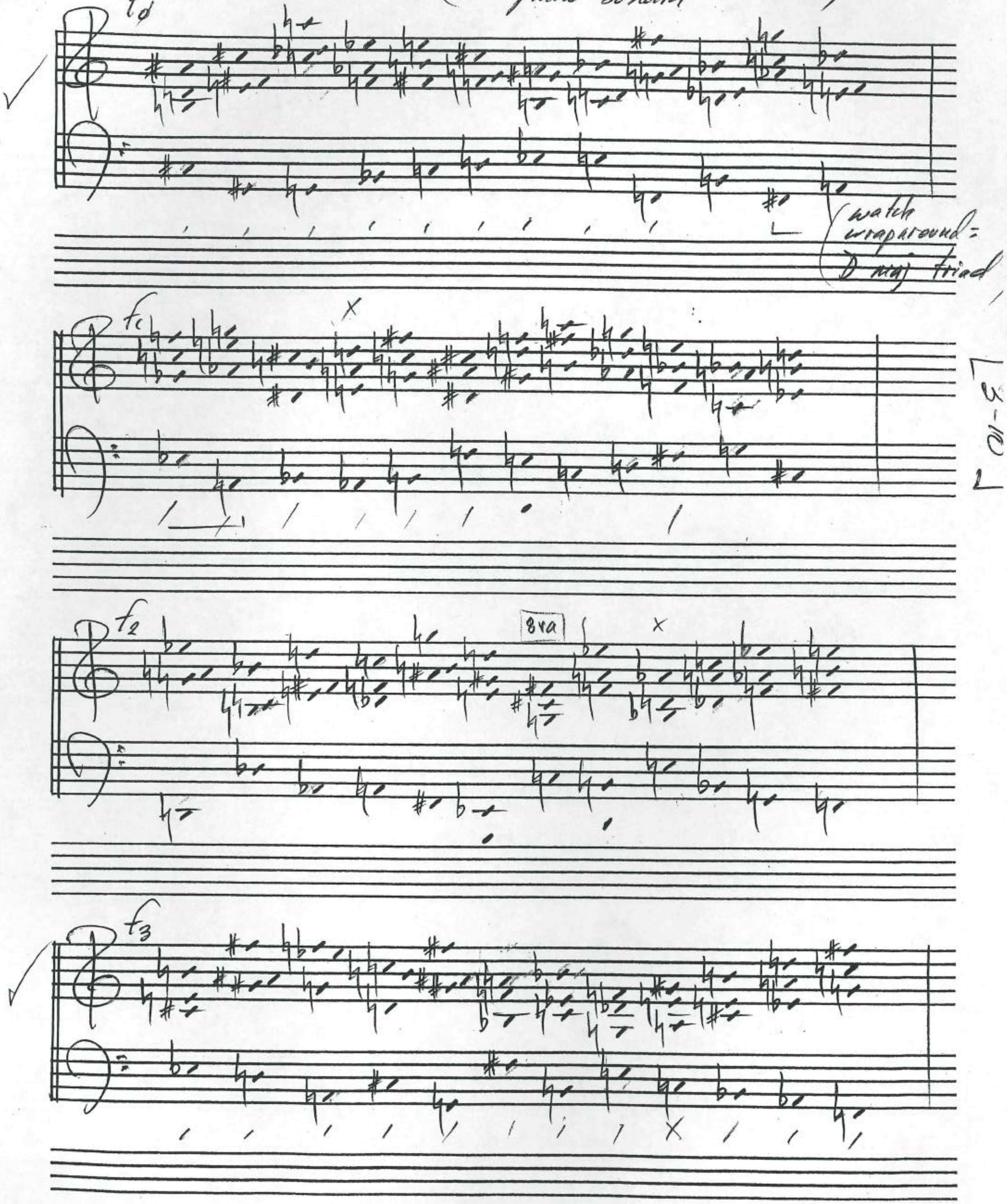
8/21/05 - ENCRYPTED Arrays (8/16/05 row) (Pg. 2)



[D-3]

8/23/05 - Encrypted arrays (8/16/05 Blues row) (pg. 3

- The whole purpose of an algorithmic system is to generate materials that might not present themselves any other way...

8/23/05 - ENCRYPTED Arrays WITH REVISIONS (pg. 1)
(SAVE 5-6 of these for piano sonata)

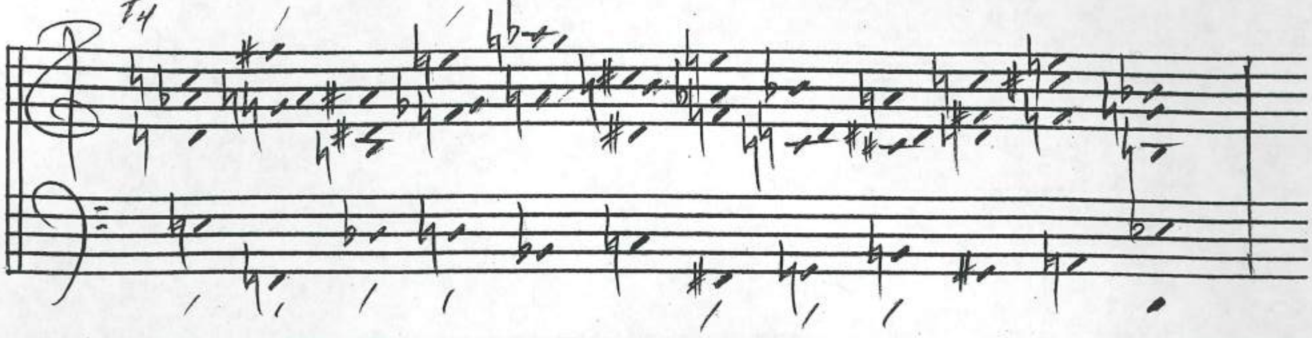


watch wraparound = D maj triad

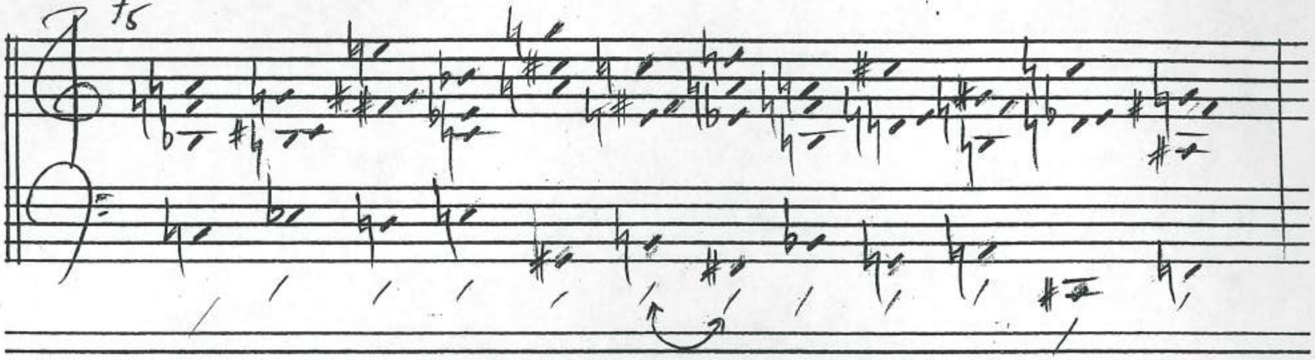
[3-10]

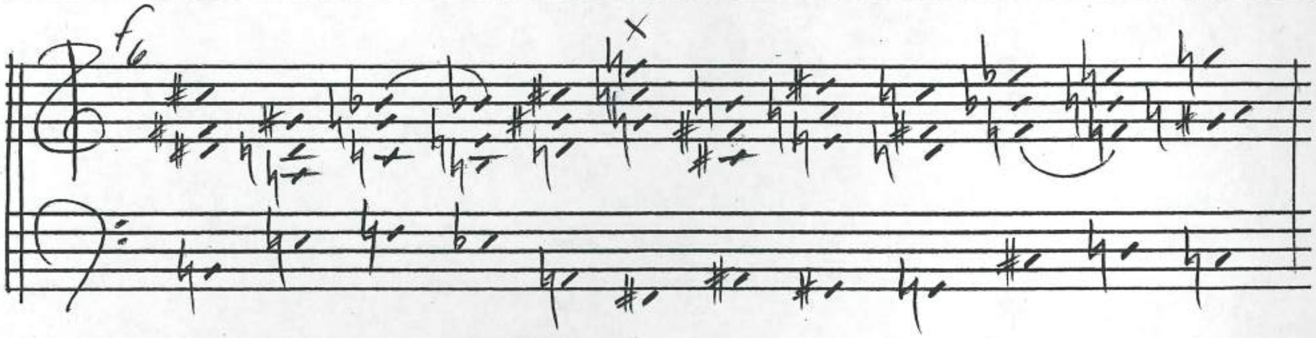
8/23/05- ENCRYPTED Arrays WITH REVISIONS (pg. 2)

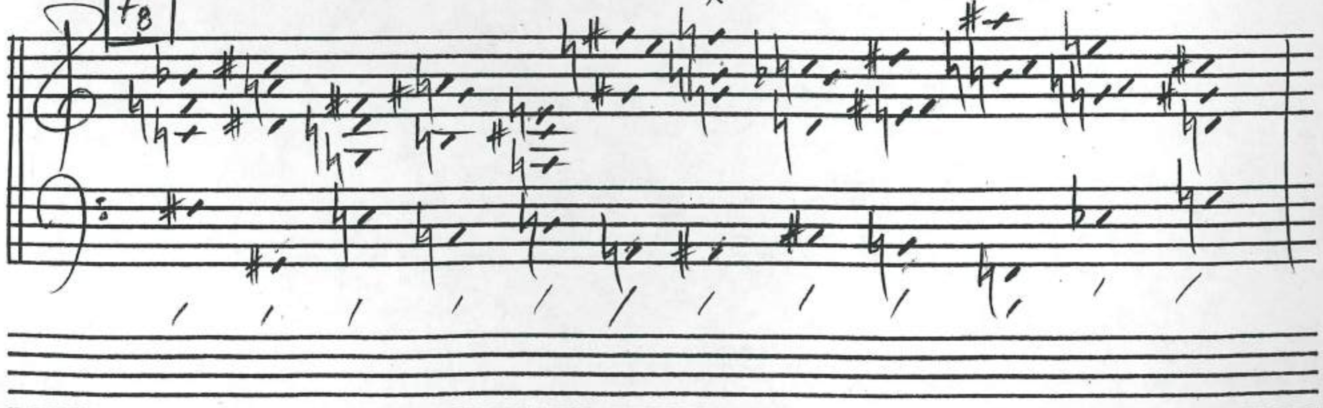
f4

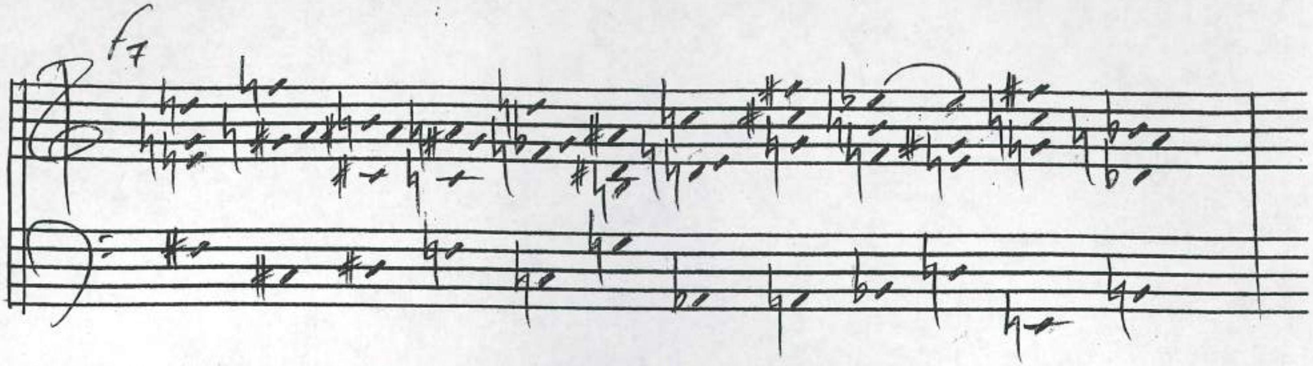


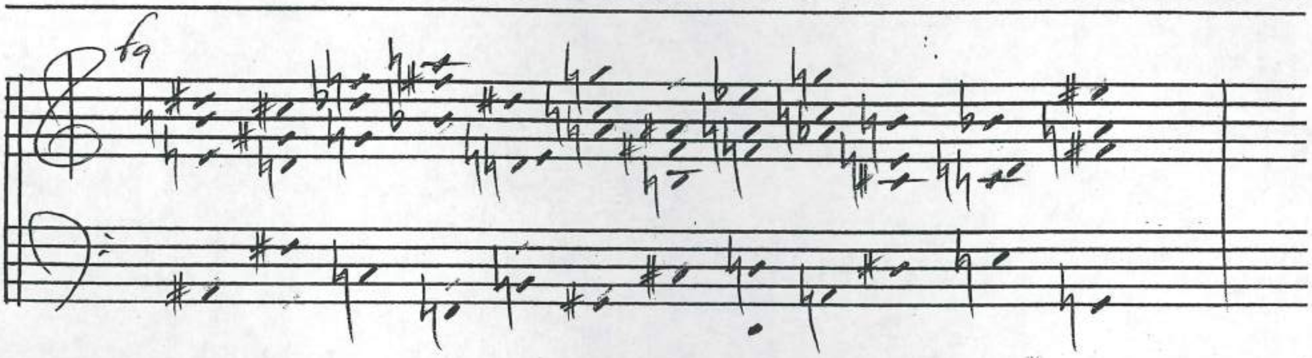
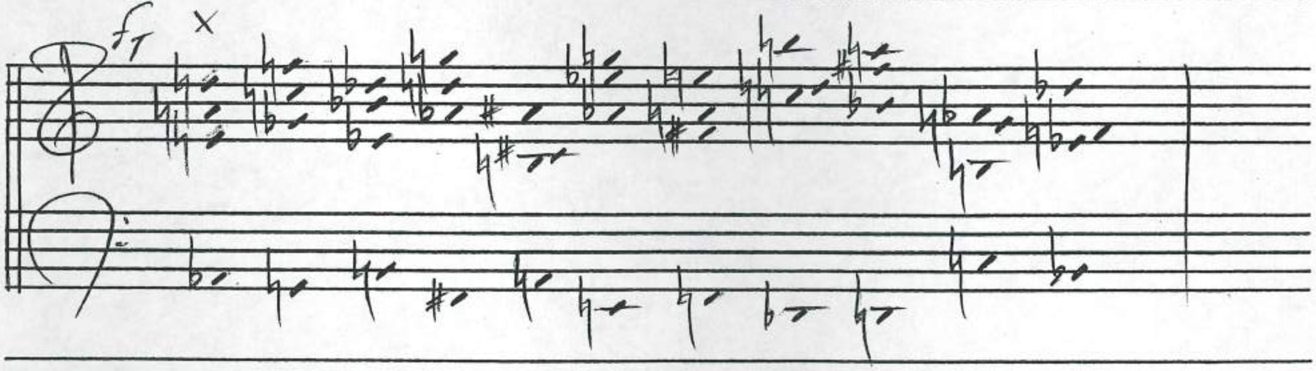
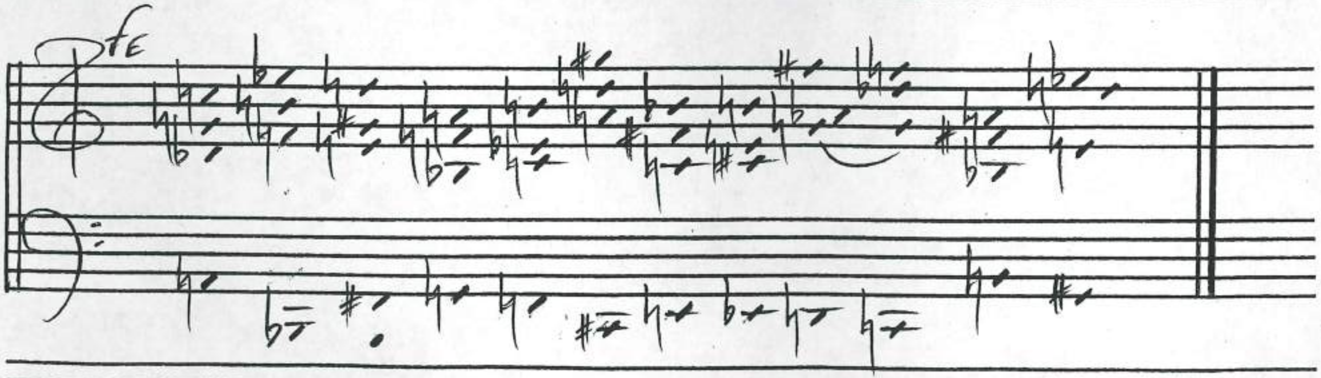
f5



f6

f8

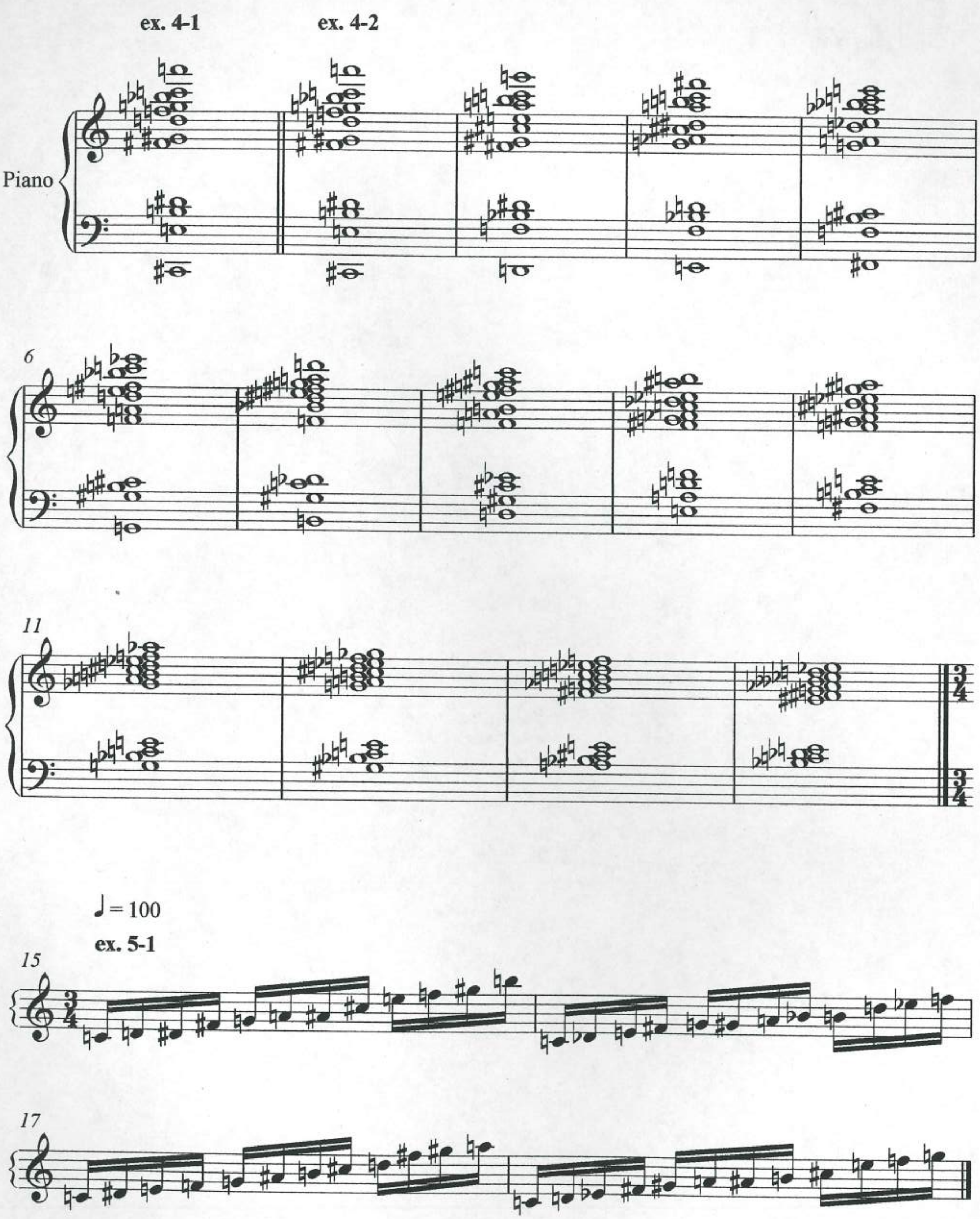


8/23/05 - ENCRYPTED Arrays WITH REVISIONS (pg. 3)

[8-10]

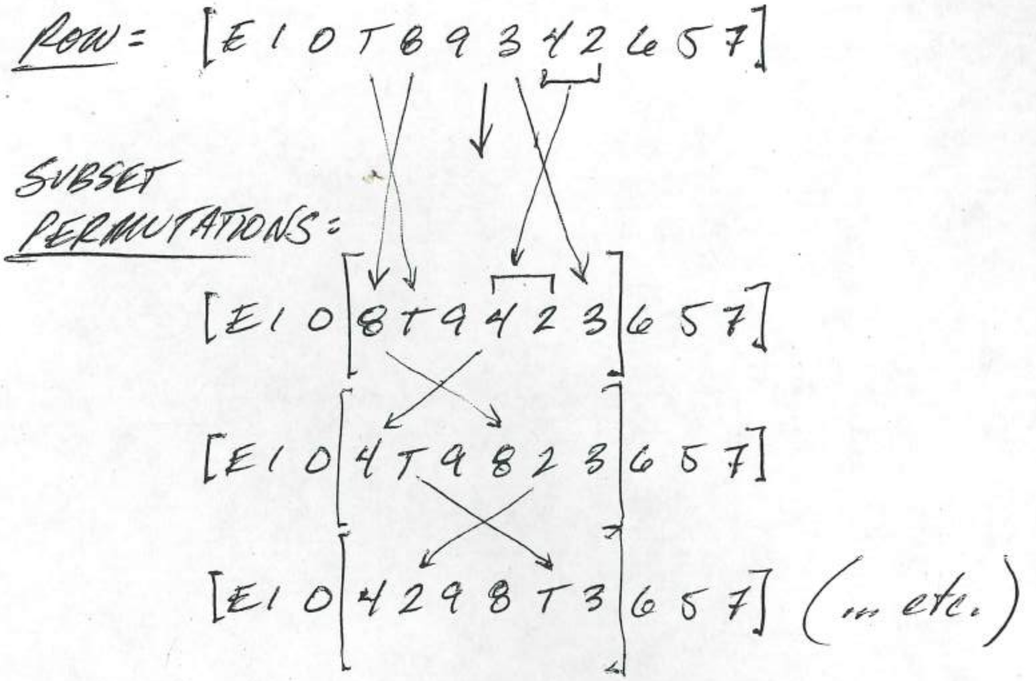


Appendix 1

Musical Examples (cont.'d)

David Shere

2/8/06 - Index subset permutations of 12-tone rows

[example 6-1]



Row = [E 1 0 T 8 9 3 4 2 6 5 7]

SUBSET PERMUTATIONS =

[E 1 0 | 8 T 9 4 2 3 | 6 5 7]

[E 1 0 | 4 T 9 8 2 3 | 6 5 7]

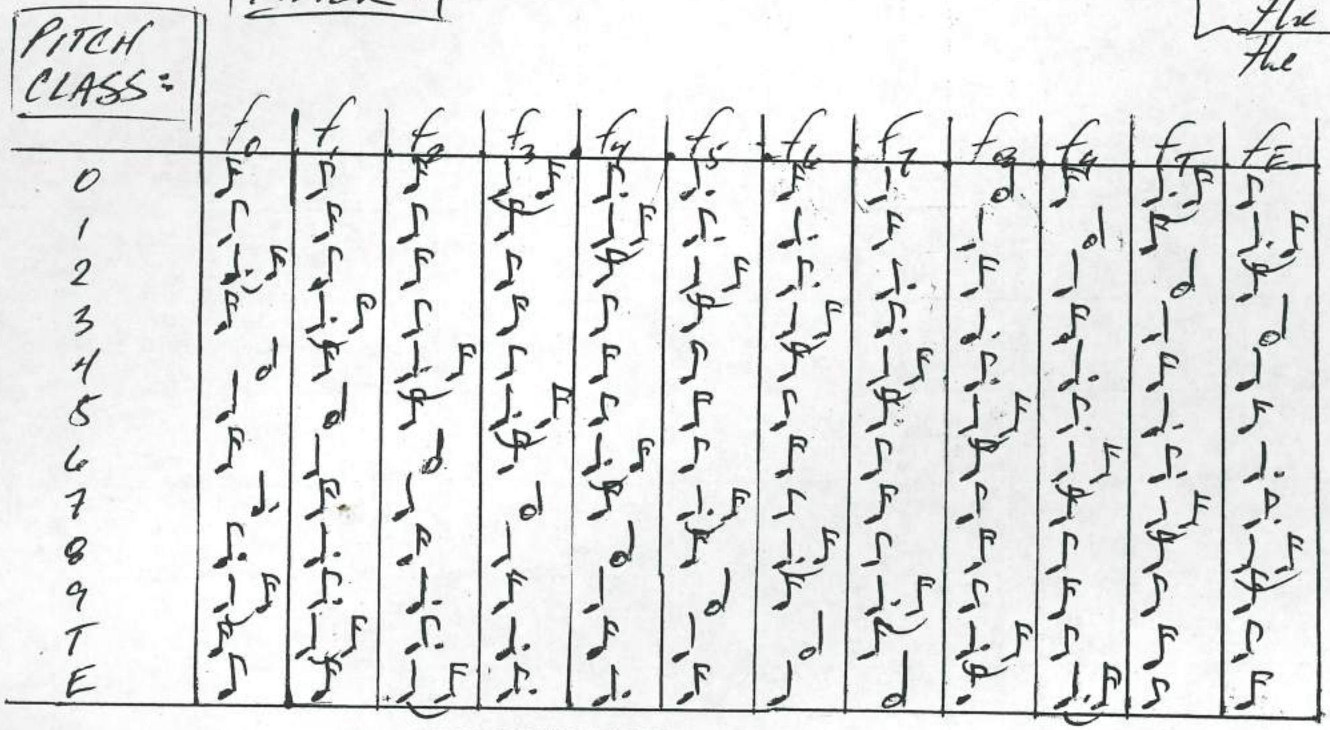
[E 1 0 | 4 2 9 8 T 3 | 6 5 7] (... etc.)

- this technique is based on a *partial* index encryption, rather than a *complete* index encryption.

— David W. Shore

2/19/05 - RHYTHM relation table

FILTER:



NOTE DURATION:

- If a pitch class encounters another iteration of the same p.c. = map the following it. onto the ~~previous iteration and replace the 1st iteration with a rest = to its remaining duration~~

1) Convert the following iteration to a rest, OR

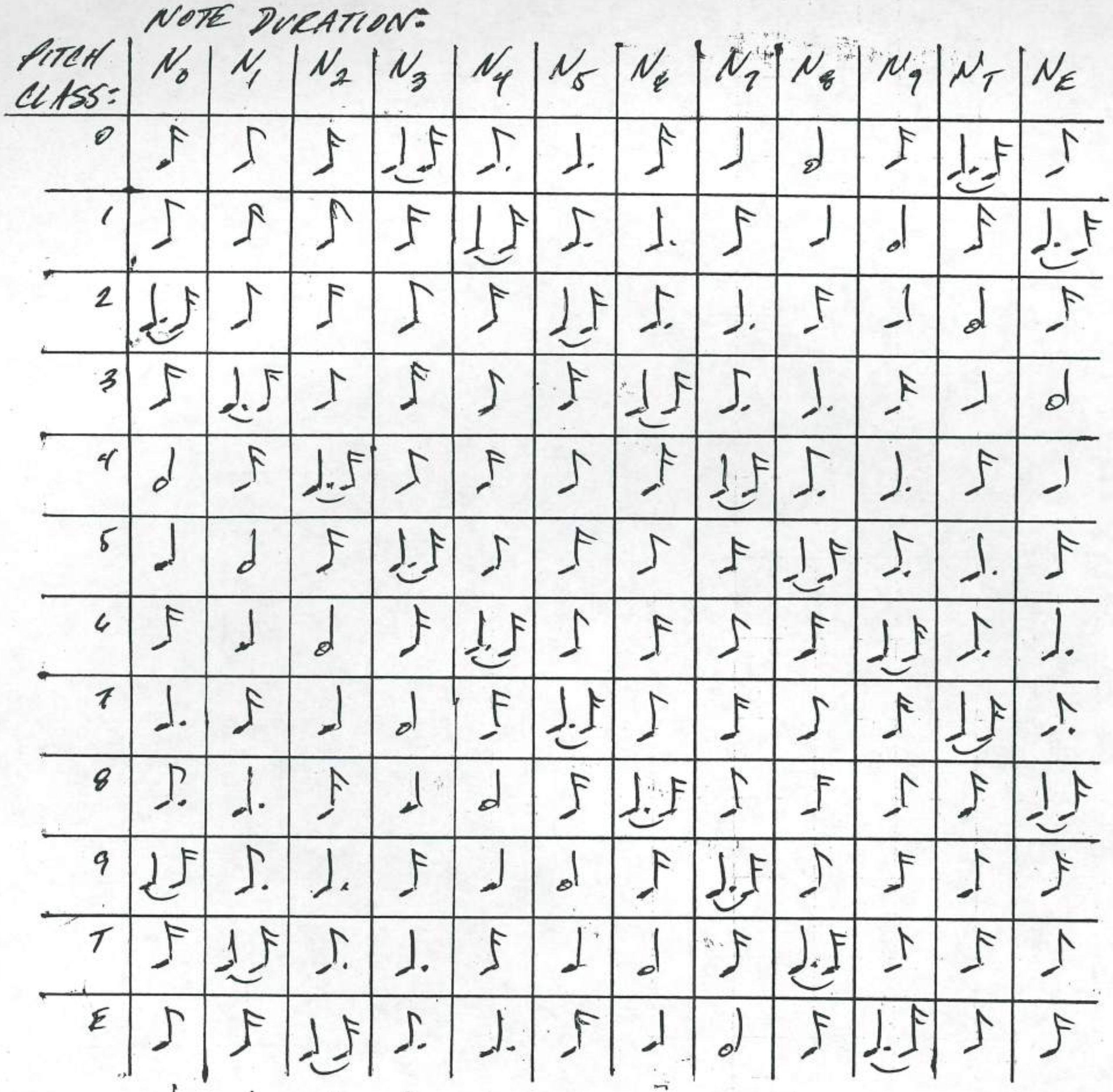
2) Convert the rhythmic space occupied by the following iteration to a prolongation of the preceding pitch-class

Choices:

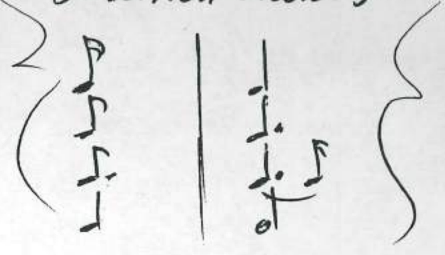
[example 7-1]

[example 7-2]

7/9/68 – RHYTHM PATTERNING TABLE I (Melodic)



Duration choices:



IN THE EVENT of PITCH CLASS overlaps:

1) Convert the subsequent iteration to rests OR

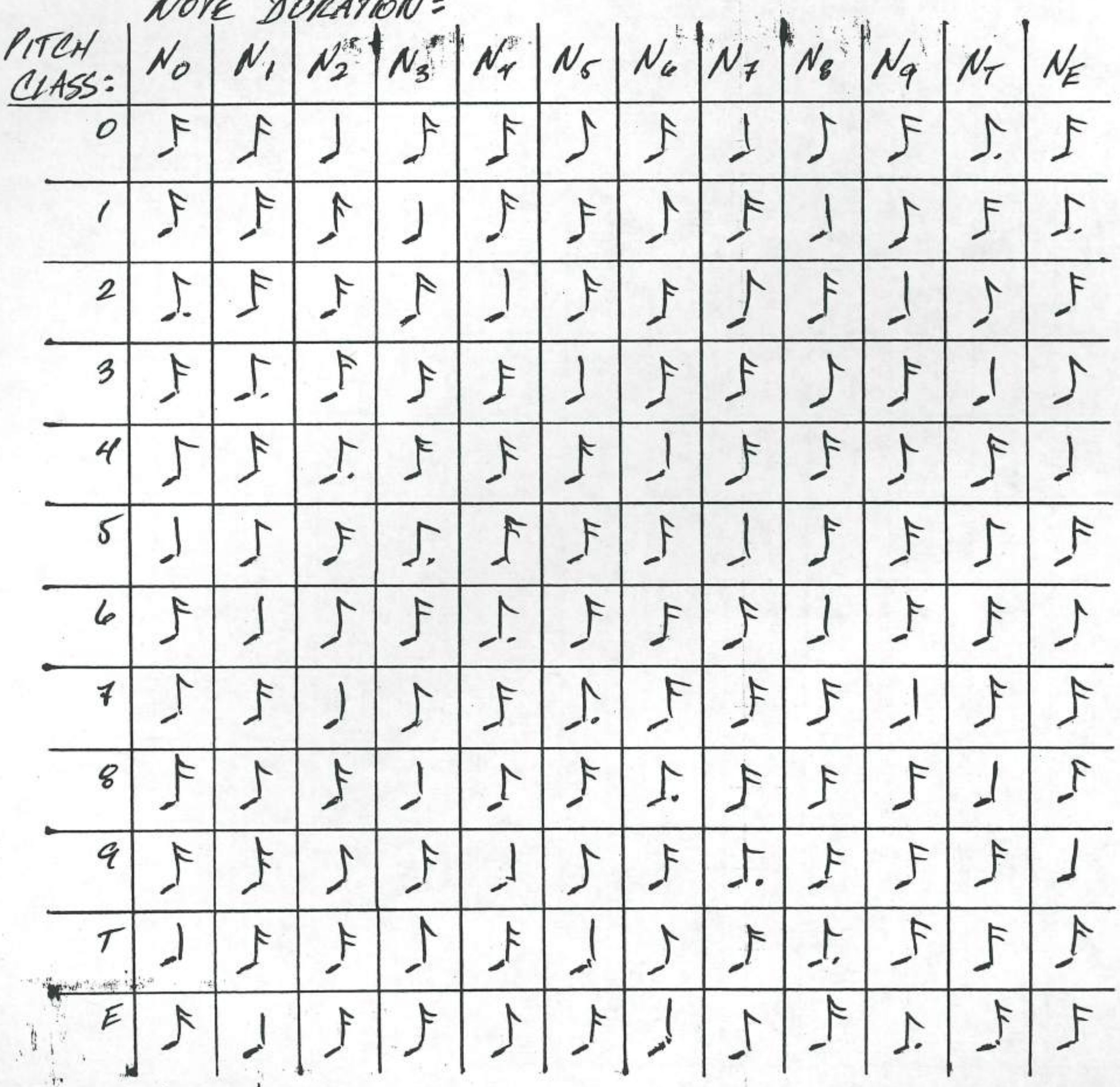
2) Convert the rhythmic space occupied by the following iteration to a prolongation of the preceding pitch-class

ALSO: any portion of a rhythmic duration may be converted to a rest

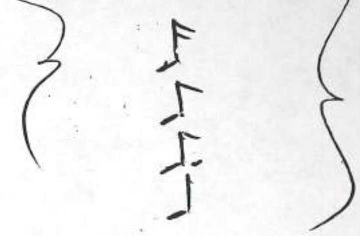
- And rests may be added arbitrarily

[ex. 7-7]

2/20/05 - RHYTHMIC FILTERING TABLE II (motor rhythm)



Durations:



USE Poly-groups:

$\overset{\frown}{4} + \overset{\frown}{5}$
$\overset{\frown}{6} + \overset{\frown}{5}$
$\overset{\frown}{7} + \overset{\frown}{6}$
$\overset{\frown}{4} + \overset{\frown}{3}$
$\overset{\frown}{7} + \overset{\frown}{4}$
$\overset{\frown}{6} + \overset{\frown}{4}$
$\overset{\frown}{5} + \overset{\frown}{3}$

2/20/05 - RHYTHMIC FILTERING TABLE III (Chorale)

″

Durations:

{ ♩ ♩.
𝅗𝅥
𝅗𝅥.
𝅝 }

	NOTE DURATIONS:											
PITCH CLASS=	N_0	N_1	N_2	N_3	N_4	N_5	N_6	N_7	N_8	N_9	N_T	N_E
0	𝅗𝅥	𝅗𝅥.	𝅗𝅥	♩.	𝅝	♩.	𝅗𝅥	𝅗𝅥.	♩	𝅝	𝅗𝅥	𝅗𝅥.
1	𝅗𝅥.	𝅗𝅥	𝅗𝅥.	𝅗𝅥	♩.	𝅝	♩.	𝅗𝅥	𝅗𝅥.	♩	𝅝	𝅗𝅥
2	𝅗𝅥	𝅗𝅥.	𝅗𝅥	𝅗𝅥.	𝅗𝅥	♩.	𝅝	♩.	𝅗𝅥	𝅗𝅥.	♩	𝅝
3	𝅝	𝅗𝅥	𝅗𝅥.	𝅗𝅥	𝅗𝅥.	𝅗𝅥	♩.	𝅝	♩.	𝅗𝅥	𝅗𝅥.	♩
4	♩	𝅝	𝅗𝅥	𝅗𝅥.	𝅗𝅥	𝅗𝅥.	𝅗𝅥	♩.	𝅝	♩.	𝅗𝅥	𝅗𝅥.
5	𝅗𝅥.	♩	𝅝	𝅗𝅥	𝅗𝅥.	𝅗𝅥	𝅗𝅥.	𝅗𝅥	♩.	𝅝	♩.	𝅗𝅥
6	𝅗𝅥	𝅗𝅥.	♩	𝅝	𝅗𝅥	𝅗𝅥.	𝅗𝅥	𝅗𝅥.	𝅗𝅥	♩.	𝅝	♩.
7	♩.	𝅗𝅥	𝅗𝅥.	♩	𝅝	𝅗𝅥	𝅗𝅥.	𝅗𝅥	𝅗𝅥.	𝅗𝅥	♩.	𝅝
8	𝅝	♩.	𝅗𝅥	𝅗𝅥.	♩	𝅝	𝅗𝅥	𝅗𝅥.	𝅗𝅥	𝅗𝅥.	𝅗𝅥	♩.
9	♩.	𝅝	♩.	𝅗𝅥	𝅗𝅥.	♩	𝅝	𝅗𝅥	𝅗𝅥.	𝅗𝅥	𝅗𝅥.	𝅗𝅥
T	𝅗𝅥	♩.	𝅝	♩.	𝅗𝅥	𝅗𝅥.	♩	𝅝	𝅗𝅥	𝅗𝅥.	𝅗𝅥	𝅗𝅥.
E	𝅗𝅥.	𝅗𝅥	♩.	𝅝	♩.	𝅗𝅥	𝅗𝅥.	♩	𝅝	𝅗𝅥	𝅗𝅥.	𝅗𝅥

[ex. 7-2]

APPENDIX 2: BIBLIOGRAPHY (Recommended Reading)

Aldwell, Edward & Carl Shachter. **Harmony and Voice-leading: 3rd Edition.** Wadsworth Group/Thomson Learning: Belmont CA, 2003.

Benjamin, Thomas, Michael Horvit and Robert Nelson. **Techniques and Materials of Tonal Music, With an Introduction to Twentieth-Century Techniques: Fourth Edition.** Wadsworth, Inc.: Belmont CA, 1992.

Chung , Ki Hyang. **Integral Serialism and the Rise of the International Avant-garde.** 1/31/06. <http://www.usc.edu/dept/polish_music/578/aug05.html>

Jay, Stephen, Donald J. Rauscher and Justine Shir-Clif. **Chromatic Harmony.** The Free Press/MacMillan: New York NY, 1965.

Marshall, Wolf. **Joe Satriani: Surfing with the Alien.** "Introduction." Cherry Lane Music Company, Inc.: Port Chester NY, 1987.

Slonimsky, Nicholaus. **Thesaurus of Scales and Melodic Patterns.** Schirmer Books/Amsco Publications: New York NY, 2003.

Straus, Joseph N. **Introduction to Post-Tonal Theory: Second Edition.** Prentice-Hall, Inc.: Upper Saddle River NJ, 2000.